Online Calibration of the EMG to Force Relationship

Christian Fleischer, Konstantin Kondak, Christian Reinicke, Günter Hommel

Technische Universität Berlin, Germany,
Institut für Technische Informatik und ME,
Einsteinufer 17, Sekr.: EN10, 10587 Berlin
{fleischer,kondak,reinicke,hommel}@cs.tu-berlin.de

***Abstract*— In this paper we present a method to calibrate the surface EMG signal-to-force-relationship online. For this, a simple biomechanical model composed of bones and muscles is used. The calibration is based on an online optimization algorithm where the error between the movement of the human and the movement computed with the biomechanical model is minimized. The proposed method will be part of a control system for an exoskeleton robot that should aid the wearer in everyday-life situations like walking, standing up and sitting down.**

In contrast to existing methods for the calculation of the EMG signal-to-force-relationship, we are not interested in the exact force values of every single muscle, but our model groups some muscles together and uses the EMG signal of one of those muscles as a representative for the group to simplify calculations.

The performance of the presented method was investigated on the leg movement in sagittal plane without contact to the environment.

I. Introduction

For many decades surface electromyography has been studied by many researchers in the medical and biomechanical fields to get a better understanding of how muscles work internally and how and when they are activated.

In recent years more and more studies have explored the relationship between single muscles and the complex movements of the human body.

Most of these studies were focused on analysing disabilites, anomalies and how to track a progress in rehabilitation. In contrast to that, only a few publications focused on using electromyographical signals in real-time to control biomechanical robots e.g. [1]–[3].

The main reason for this is the difficulty to map the EMG signal into the force a muscle is producing [4]. The approximated relationship itself is not too complex, see e.g. [5], but influenced by many different parameters. Some of these parameters only vary among different subjects, but some are even different from day to day. Examples for the first type are: point of origin and insertion of every muscle, body weight and lengths of bones and tendons for example. Some representatives for the other class of factors are moisture on the skin, fatigue of the muscles and blood circulation.

A. Why EMG?

There are many different approaches for tracking movement of human beings. Some use ultrasonic or visual sensors, some goniometers, accelerometers or other techniques. Every technique has its advantages and disadvantages. All those systems sense the current movement. With EMG signals it is possible to track the *intended* movement, which might differ from the current movement due to obstacles or lack of sufficient muscle-power. And if the EMG sensors are placed carefully, the intention should even be available ahead of time. The reason for this is that EMG signals are detectable slightly before the actual movement is performed, because muscles take some time to produce the force after having received the activation signal.

Those properties of EMG signals should be very helpful in developing a real-time control unit for an exoskeleton.

B. Goal

More or less uncalibrated parameters for the EMG-to-force relationship could be used with an experimentally identified threshold to activate certain patterns of movement [1], [2]. But the idea of our work is to include the EMG signals directly into the control loop of the exoskeleton to allow for more adaptable and spontaneous movements. To achieve this, an automatic calibration method for those parameters is required. The more accurate the relationship between the EMG signals and the forces exerted by the muscles is identified the more accurate the system attached to the human body can be controlled and the interaction will be more natural. The same control scheme and calibration can also be used with haptic, remotely controlled or similar devices.

The paper is organized as follows: In sec. II the biomechanical model, the EMG set up and the motion capturing system are described. In sec. III the optimization algorithm is explained in detail and in sec. IV the properties of the suggested approach are summarized. The experimental setup is described in sec. V. Results demonstrating the performance of the calibration are presented in sec. VI and discussed in sec. VII.

II. Environment

In this section the environment is described in which the calibration system is embedded. Please refer to Fig. 1 for an initial overview of the whole system.

As stated before, the basic idea is to let the *Human* interact with the *Mechanical System*. To achieve this, the

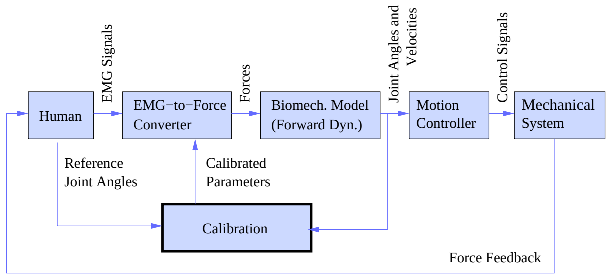

Fig. 1. General control scheme for a mechanical system attached to the human body: The EMG signals from the subject are captured, converted into forces and brought into the biomechanical model. The model then calculates the desired joint angles and velocities and passes them to the motion controller that is controlling the mechanical system. The calibration compares the reference angles in the joints with the angles computed by the model and modifies the EMG-to-force-parameters for the muscles.

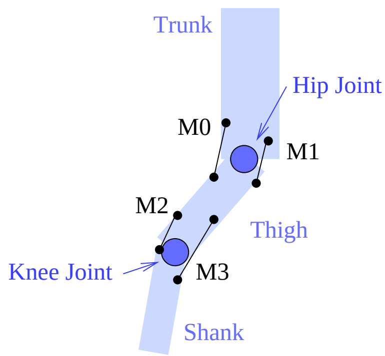

Fig. 2. The biomechanical model composed of trunk, thigh, shank and four muscles: $M0$, $M1$, $M2$, $M3$

EMG Signals are collected from some of the muscles of the subject and converted to forces in the block *EMG-to-Force Converter*. The resulting *Forces* are then fed into the *Biomechanical Model* that describes the human subject wearing the sensors. Through forward dynamics computation accelerations for all body parts are calculated from the muscle forces and the current state of the model. After double-integrating those accelerations, the resulting *Kinematical Data* (joint angles and velocities) are interpreted as the desired movement of the human. The *Motion Controller* takes this desired movement, modifies it if it does not lead to a stable pose of the human and computes the control signals for the *Mechanical System*. Because of the connection between the human body and the mechanical system, the motion of the actuators of the mechanical system affect the human body (*Force Feedback*).

To be able to calibrate the parameters of the *EMG-to-Force Converter* the calculated *Kinematic Data* from the *Biomechanical Model* are compared with the *Reference Kinematic Data* taken from the human body with additional sensors (see sec. II-C). The computed parameters are then brought into the *EMG-to-Force Converter* again. Those additional sensors should be attached to the human with or without an exoskeleton to allow recalibration whenever necessary.

The block *Calibration* is the main subject of this work. The calibration is implemented as an online optimization process of the EMG-to-force relationship parameters by minimizing the quadratic difference between the desired and reference movement data. Here only the movement of one leg in sagittal plane without interaction with the environment is considered (refer to sec. V for the experimental setup).

A. *Biomechanical Model and EMG-to-Force Converter*

In literature a lot of information can be found about the anatomy of muscles, joints, tendons and tissues, see e.g. [6]. The properties of those are well investigated and available. Also elaborated investigations have been done on biomechanical modelling and verification of different models for muscles, tendons and joints, see e.g. [7]. Unfortunately, with increasing complexity of the model, more parameters have to be calibrated - albeit most of them not in real-time and for each subject only once. But nevertheless, it is advisable to keep the model as simple as possible. One reason for this is the fact, that it is impossible to get EMG signals from every muscle and in equally good quality. So even though it is possible to create a complex and realistic model of the lower extremities, see e.g. [8], it is not possible to provide all those muscles in the model with properly recorded EMG values. This would result in an uncompensated defect of some muscles.

For our experiments we have chosen a very simple model: It consists of the trunk, thigh, shank and four muscles, as shown in Fig. 2. Please note that each of those muscles in our model is representing a collection of muscles in the human body and does not necessarily have anatomical analogy. Of course the EMG signals are acquired from specific muscles of the human. For this we selected the muscles by their contribution to the most important movements of the leg: m. sartorius ($M0$), m. glutaeus maximus ($M1$), m. quadriceps femoris ($M2$), and m. semimembranosus ($M3$). We are aware that this is a rough approximation, but keeping in mind the goal to provide input values for control units of a mechanical construction and not pursueing clinical analysis, this seems to be an adequate simplification. If required, more muscles can easily be integrated into the model.

Each of those muscles is only represented by a contractile element. The passive elastic and viscous elements are summed up in the joints spanned by the muscle (in contrast to the Hill model) together with the friction of the joints. This simplification is similar to one presented in [9]. The overall friction is assumed to be $5.0\frac{Nms}{rad}$ in the hip joint and $0.2\frac{Nms}{rad}$ in the knee. The knee value is inspired by [10] and [11], the value for the hip was optimized by hand. So the parameters of one muscle in our model are: the point of origin and insertion and the two parameters in the EMG-to-force function:

$$F_{EMG}(\phi_s, \phi_e) = \phi_s \left(1 - e^{-\phi_e x(t)}\right) \tag{1}$$

where $x(t)$ is the postprocessed EMG signal and ϕ_s, ϕ_e two muscle parameters. Since muscle origin and insertion do not vary for the same average adult, offline optimization can be performed here. But in most cases even automatic calculation derived from the body dimensions should suffice. Due to lack of space here, the biomechanical model can be received from the authors on request. There are many other properties described in literature that are neglected so far. But they do not have an influence on the calibration algorithm itself. They can be incorporated later if needed.

The thigh and shank are modelled as cylindrical rigid bodies. The trunk was assumed to be fixed. Body masses for the thigh and shank are calculated as fixed fractions of the total body weight ($m_{total} = 88kg$) of the subject (the figures can be found e.g. in [12]). The dynamic equations were derived using Kane's formalism [13], and have the following form:

$$\mathbf{M}(\mathbf{q})\,\dot{\mathbf{u}} = \mathbf{f}(\mathbf{q},\mathbf{u}) + \mathbf{T}(\mathbf{q},\mathbf{u}) \tag{2}$$

where

- $\mathbf{q} = (q_1,\, q_2)^T$ is the vector of generalized coordinates, which are angles in knee and hip joints
- $\mathbf{u} = (u_1,\, u_2)^T$ is the vector of corresponding generalized velocities (with $\dot{\mathbf{q}} = \mathbf{u}$, the dot denotes the time derivative in a Newtonian reference frame)
- $\mathbf{M}(\mathbf{q})$ (matrix function) takes into account the mass distribution
- $\mathbf{f}(\mathbf{q},\mathbf{u})$ (vector function) describes the influence of both inertial forces and gravity
- $\mathbf{T}$ is a vector of the generalized forces applied to the system. For the model considered, these are:
  - the forces produced by the muscles (F_{EMG}) multiplied with the nonlinear function $\mathbf{g}(\mathbf{q})$ (current system configuration and geometry of the muscles)
  - friction torques in joints (depending on $\mathbf{u}$)

The dynamic eqs. (2) were generated with the symbolic manipulation tool AUTOLEV [14]. The script for the system description and eqs. generation can be received on request.

B. EMG Setup

The EMG signals are sampled with 1 KHz from DelSys 2.3 Differential Signal Conditioning Electrodes [15] with an inbuilt gain of $1000\frac{V}{V}$ and a bandpass filter from 20-450 Hz. The input data is rectified and then smoothed by a lowpass filter with a cutoff frequency of 5 Hz [16].

C. Reference System

The reference system is needed to capture the actual movement of the human limbs for the calibration step. We have used a reference system based on the two axis accelerometers ADXL210 from AnalogDevices Inc. [17].

As shown in Fig. 3 the orientation of the limb in the sagittal plane can be calculated by projecting the earth gravity field into the x- and y-axis of the sensor:

$$q_i = \arctan 2\left(\frac{G_{y\,i}}{G_{x\,i}}\right)$$

One sensor was placed on the thigh and one on the shank, both as close as possible to the rotation axes of the joints to reduce the inertial acceleration resulting from limb movement (the error is small enough below $45\frac{deg}{s^2}$). Due to the nature of the sensors, there is only 0.5% temperature drift and a peak-to-peak noise below 2%. Unfortunately, only 10% of the full range of the sensor can be utilized (full range is $\pm 10g$).

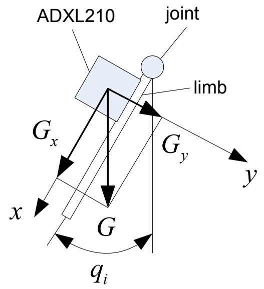

Fig. 3. Capturing the limb movement with two axes accelerometer ADXL210.

III. OPTIMIZATION

The overall goal of the optimization algorithm is to allow online calibration of crucial parameters of the system with a small number of movements and automatic recalibration if a sufficient number of new measurements are available to adapt the parameters in a potentially different situation (e.g. muscle fatigue, moisture on skin).

As already mentioned in section II-A the most varying parameters are ϕ_s and ϕ_e. Together they form a parameter set $P_m = \{\phi_s, \phi_e\}$ for the muscle m.

A straightforward approach would be to record a history over a certain period of time and perform some error minimization between the angles of the reference system and the calculated angle output of the model while modifying all parameters of all muscles.

Beside the immense computation time needed for a complete recalculation of the whole movement history for one optimization step (even when reducing the data to every n-th value or the last t seconds), this method has a major drawback: It does not take into account the activation of a muscle. Parameters of a muscle should only be optimized when a certain amount of representative and different EMG values have been collected. It does not make sense to calibrate parameters contributing to a force-function (and force leads to acceleration), without motion to get reference values.

To avoid this problem the algorithm collects measurements in a table with the size of T entries. Each entry $E_n, 0 \le n < T$ in the table contains the angles $q_1(t)$, $q_2(t)$ and angular velocities $u_1(t)$, $u_2(t)$ of the sensors from the reference system at $t = t_n - \Delta t_n$ and $t = t_n$ and the timestep Δt_n (between the last measurement and the current measurement, Fig. 4).

Calculation of the entry index in the table is based on the idea of hashtables: We define a function

$$h(x) = x\frac{T}{\alpha} \quad \text{with} \quad x > 0,\ \alpha > 0$$

that calculates the index from the postprocessed EMG signal $x(t)$. To emphasize a certain working range of

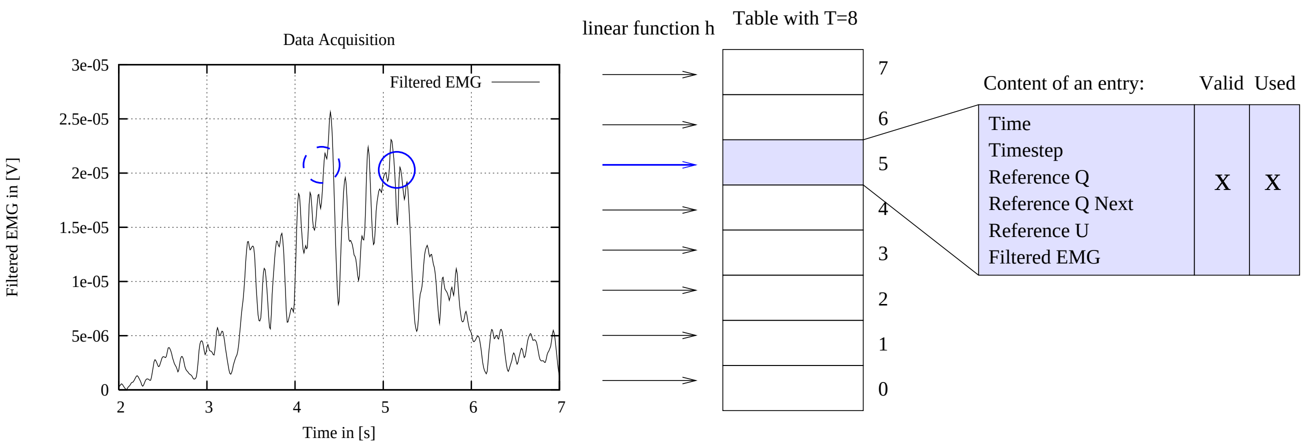

Fig. 4. The flow chart shows the data processing after low pass filtering the rectified EMG values: The example table has a size of 8 entries (realistic sizes start at 20-30) and is indexed by the linear function $h(x)$ as shown in the middle. The right part shows the information stored for every measurement in the table. The solid circle in the left part of the diagram represents the current table entry, the dashed circle was the first measurement appearing in the table. This entry has been overwritten several times afterwards.

the muscle a more complex distribution can be chosen. The function $h(x)$ is allowed to exceed the upper table boundary $T-1$. In this case, the table is resized so that $h(x) = T-1$ is fulfilled. By doing this, α can be a rough estimation of the maximum EMG signal and does not need to be the upper limit. All this ensures that the table contains values from different levels of activation of the muscle. Longer periods of inactivity or *unchanging* activity will not affect previously collected measurements except for entries with the same level of activity.

When the table has at least $n_{min} = 0.5 \times T$ number of entries, the parameter optimization is started. The mathematical optimization of the parameters is performed with the *Nelder-Mead Simplex Algorithm* [18]. This algorithm requires an error-function that is evaluated several times during a single step of the algorithm and therefore should be executed as fast as possible to allow online optimization. Evaluating the error-function is performed by calculating the biomechanical model for a single timestep Δt_n for every valid entry E_n in the table. The error $e(P_m)$ for a single timestep t_n of the table entry E_n and the parameter set P_m is

$$e(P_m) = \sum_i \{q_i^{model}(t_n) - q_i^{ref}(t_n)\}^2$$

where i is the index of every joint that is affected by the muscle. The total error of the current calibration step is

$$E(P_m) = \sum_{t_n} e(P_m)$$

where t_n refers to all measurements in the table.

The Nelder-Mead algorithm terminates when either a local minimum criterion is met (refer to [18], [19] for more details of the method) or a maximum number of iterations is reached. Then all entries in the table become tagged as *used*. A recalibration is started after $n_{new} = 0.75 \times n$ table entries have been added or overwritten since the last calibration.

The optimized parameters are immediately brought into the biomechanical model and are used for all further calculations. If the newly emerged F_{EMG} function differs

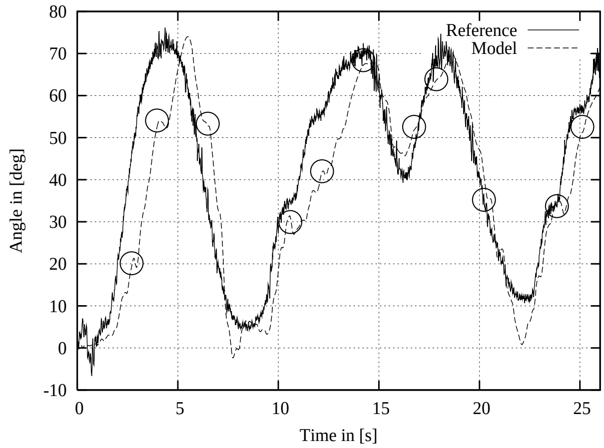

Fig. 5. The diagram shows the lifting of the thigh in an arbitrary pattern (standing position, thigh angle in sagittal plane, $0deg$ means perpendicular to the ground) with online calibration and evaluation. The solid curve is the reference measurement, the dashed one is the curve from the model. Points in time where calibration was performed are marked with circles. Note the immediate reaction of the model to the new parameters. Sometimes this reaction might lead to unwanted overshootings for large changes. The initial table size was 30 but during the process it was enlarged to 36.

from the old one significantly, the model might oscillate because of the resulting acceleration. This can be avoided by overwriting the generalized coordinates q_j and velocities u_j with the values calculated from the reference system if needed.

IV. Summary of the Important Properties

The method has the following properties:

- Only one muscle is calibrated at the same time, but a single measurement can be put into any number of tables.
- The smaller the number of table entries, the faster the method is.
- The larger the number of entries in the table, the more accurate the optimization will be.
- Recalibration is only performed if a minimal number of new, representative measurements have been recorded.

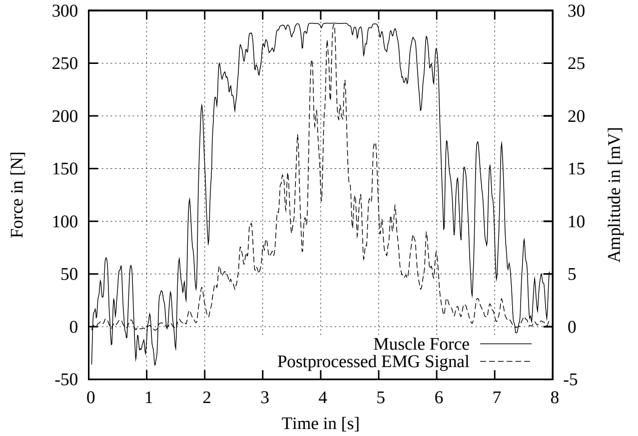

Fig. 6. Fig. shows the postprocessed EMG values (rectified and low pass filtered) and the resulting force values, calculated with the EMG-to-force function F_{EMG} ($\phi_s = 288.8$ and $\phi_e = 349.2$).

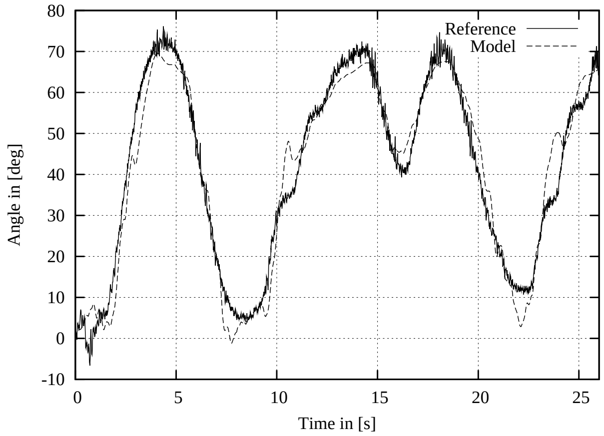

Fig. 7. The same curve as in Fig. 5, replayed and with optimization turned off. The parameters $\phi_s = 275.3$ and $\phi_e = 237.9$ used for F_{EMG} have been calculated with online optimization in the original run shown in Fig. 5. As could be expected, the curve-fitting is more accurate, since from the beginning good parameters are known and the model is not disturbed by parameter feedback.

- A single measurement entry has a timestep of $\Delta t_n = \frac{20}{samplerate}$ to avoid artefacts in the EMG signal and noise in the reference system.

V. Experiments

The experiments were performed in an upright standing position of the subject. The left foot was put on the ground for support, the right thigh was lifted with different velocities in sagittal plane with the shank pointing down. The torso and hip was held in position. In the biomechanical model, the hip is fixed in place so that the supporting leg does not need to be modelled.

During the raising and lowering of the leg, the main force contribution from the muscles come from the group represented by $M0$ (all other groups can be neglected in our experiments) and this is the only group considered in the diagrams here. The displayed angle is the angle between the shank and the line of gravity: 0 degrees means pointing down to the ground, 90 degrees is parallel.

The movement was not hindered by any obstacles and the only external force affecting the leg was gravity. If other forces are applied to the leg (like ground reaction forces) they have to be added to the biomechanical model as well. But that does not affect the calibration algorithm itself.

In Fig. 5 the results of an experiment are shown: The reference angle of the thigh is compared to the angle calculated with the biomechanical model during online calibration. The Fig. 6 shows the EMG signal together with the corresponding force for the first 8 seconds of the movement.

VI. Results

As can be seen in Figs. 5 and 7, the calibration method produces a sufficiently good relationship between the EMG signal of the muscle and the resulting force so that the correlation between the reference angle and the calculated model angle is clearly visible.

While the calibration is performed, there are sometimes unwanted rough changes when the parameters are fed back,

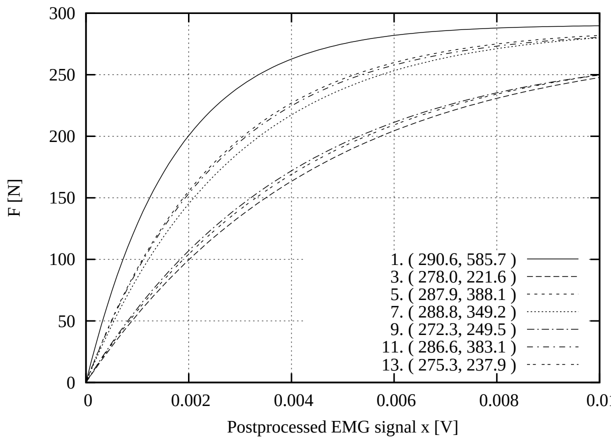

Fig. 8. This Fig. displays the F_{EMG} function with the different parameters resulting from the calibration steps marked in Fig. 5. The range of the postprocessed EMG signal is roughly between $0V$ and $0.0003V$ in this experiment.

but this could be reduced by gradually changing the old parameters to the new ones over a short period of time.

Fig. 7 shows a replay of the same movements, but with no online calibration. The parameters used here originate from the original run. This demonstrates that the model reflects the properties of the subject's leg very well - if not in every aspect, but in those that are relevant for the regarded movement.

Fig. 8 shows the F_{EMG} function with the parameters resulting from the optimizations marked in Fig. 5. The functions look quite similar (in an ideal model the parameters would always be the same) but do not converge toward a fixed value. That lies in the nature of the calibration method, because old measurements might be overwritten after a while (to minimize the computation cost) and thus do not have an effect on the calibration afterwards.

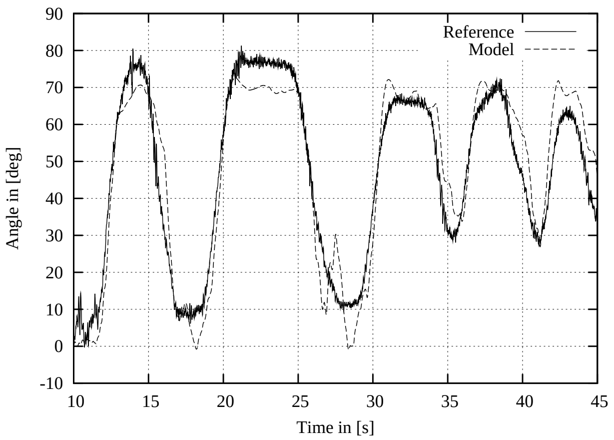

Fig. 9. During this experiment no calibration was performed. The parameters ϕ_s and ϕ_e have been taken from the prior experiment shown in Fig. 5. From the set of available values the results from calibration step 7 have been taken, since they are quite close to the average.

VII. Discussion

In section VI we have seen that the calibration method produces values that lead to quite a good imitation of the subjects's movement through the model. But what does *quite good* mean? This certainly depends on the whole application of the model. As stated in the introduction, our main focus is to allow interaction of a human being with an exoskeleton robot and to determine the intention of movement from a subject as early as possible. That means, a large delay between signal acquisition and presentation of results is to be avoided and can, if needed - within limits of course - be traded for accuracy.

As seen in Fig. 7 the delay is small if the parameters are defined well. So that raises the question, if recalibration is necessary very often. In Fig. 9 a different movement pattern (but same conditions as before) has been tracked with the model parameters from the well known experiment shown in Fig. 5. As can be seen, the results are surprisingly good. But this experiment had been performed within a few minutes after the first experiment. So the physical conditions of the subject (mainly the skin) has been more or less identical. But when the experiment was repeated another day (the sensors have been removed in-between), the parameters had to be recalibrated to lead to good results again. So it is wise to recalibrate at the begin of every experiment and after some time just in case anything has changed.

VIII. Conclusion

A method for online calibration of the EMG to force relationship was presented. The proposed algorithm was designed as part of the control system for an exoskeleton. In this context the main attention was paid not to the anatomical correctness of the biomechanical model of the human and the exact EMG-to-force relationship for single muscles but to the capability of the system to calculate the intended movement of the human.

It was shown that with the presented version of the algorithm calculation of the intended movement of the thigh is possible. At the moment, the results of the calculated movement are achieved almost at the same time the movement is performed. We are working on improvements of the algorithm, so that a reliable forecast of the human movement becomes possible. The ways to achieve that could be: choosing better parameters for muscle origin and insertion as well as adding one or two properties of the muscles to the model.

One of the next steps will be to extend the biomechanical model to allow experiments with more complicated movements where reaction forces (e.g. ground reaction force) of the environment will be incorporated.

All the experiments have been performed with healthy persons and without an exoskeleton. Tests with disabled persons have to be done yet. But we hope that in the next future it will be possible to realize an intuitive EMG-based human-to-robot interface.

References

[1] S. Lee and G. N. Sridis, “The control of a prosthetic arm by EMG pattern recognition,” *IEEE Transactions on Automatic Control*, vol. AC-29, no. 4, pp. 290–302, 1984.
[2] O. Fukuda, T. Tsuji, H. Shigeyoshi, and M. Kaneko, “An EMG controlled human supporting robot using neural network,” in *Proceedings of the IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 1999, pp. 1586–1591.
[3] S. Morita, T. Kondo, and K. Ito, “Estimation of forearm movement from EMG signal and application to prosthetic hand control,” in *Proceedings of the IEEE Int. Conf. on Robotics & Automation*, 2001, pp. 3692–3697.
[4] J. V. Basmajian and C. J. De Luca, *Muscles Alive: Their Functions Revealed by Electromyography*. Williams & Wilkins, 1985.
[5] A. L. Hof, “EMG to force processing,” Ph.D. dissertation, University of Groningen, 1980.
[6] N. Palastanga, D. Field, and R. Soames, *Anatomy and Human Movement. Structure and Function*. Butterworth Heinemann, 2002.
[7] M. Epstein and W. Herzog, *Theoretical Models of Skeletal Muscle: Biological and Mathematical Considerations*. John Wiley & Sons, Inc., 1998.
[8] 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,” *IEEE Transactions on Biomedical Engineering*, vol. 37, no. 8, pp. 757–767, Aug. 1990.
[9] R. Riener, “Neurophysiological and biomechanical modeling for the development of closed-loop neural prostheses,” Ph.D. dissertation, Technische Universität München, 1997.
[10] R. Happee, “Inverse dynamic optimization including muscular dynamics, a new simulation method applied to goal directed movements,” *Journal of Biomechanics*, vol. 27, no. 7, pp. 953–960, 1994.
[11] M. Günther, “Computersimulation zur Synthetisierung des muskulär erzeugten menschlichen Gehens unter Verwendung eines biomechanischen Mehrkörpermodells,” Ph.D. dissertation, University of Tübingen, 1997.
[12] D. A. Winter, *Biomechanics and Motor Control of Human Movement*. John Wiley & Sons, Inc., 1990.
[13] T. Kane and D. Levinson, *Dynamics OnLine: Theory and Implementation with Autolev*. Kane Dynamics, Inc., 2000.
[14] *Kane Dynamics,Inc.*, http://www.autolev.com.
[15] *DelSys, Inc.*, http://www.delsys.com.
[16] M.Thompson, M. Voigt, and M. de Zee, “Evaluation of lower extremity musculo-skeletal model using sit-to-stand movement,” in *Proceedings of the 19th congress of the International Society of Biomechanics*, 2003.
[17] *Analog Devices, Inc.*, http://www.analog-devices.com.
[18] J. A. Nelder and R. Mead, “A simplex method for function minimization,” *Computer Journal*, no. 7, pp. 308–313, 1965.
[19] F. H. Walters, L. R. Parker, S. L. Morgan, and S. N. Deming, *Sequential Simplex Optimization*. CRC Press, 1991.