Towards overcoming the bottleneck of optimizing control parameters in finite element active human body models

Oleksandr V. Martynenko1,*, Katrin Stollenmaier2, Carola A. Endler1, Fabian T. Neininger1, Syn Schmitt1, and Daniel F.B. Haeufle2

1 Institute for Modelling and Simulation of Biomechanical Systems, Stuttgart Centre for Simulation Sciences (SC SimTech), University of Stuttgart, Nobelstrasse 15, Office 007, 70569 Stuttgart, Germany
2 Multi-Level Modeling in Motor Control and Rehabilitation Robotics, University of Tuebingen, Otfried-Mueller-Strasse 25, 72076 Tuebingen, Germany

The number of Finite Element Active Human Body Model (FE aHBM) applications for the design and test of vehicle safety systems is growing. Primarily they are used for simulations of the accident pre-crash phase where the influence of occupants active movements is significant. Such models are capable of accounting for dynamic human behaviour and reflexes by incorporating bio-inspired muscle controllers. These controllers need to govern hundreds of active muscle elements during simulation in every time-step thereby dramatically increasing runtime compared to passive HBMs. As runtime is an essential element of the entire research and development process of a new vehicle, new approaches for its reduction are required. The current contribution presents methods for the tuning of controller and active muscle element parameters using a reduced multibody (MB) model with a subsequent transfer to a fully deformable FE model.

1 Introduction

FE aHBMs allow analysing the vehicle occupant’s voluntary motion during different phases of crash simulations. To make such a model biofidelic, one needs to implement musculoskeletal systems and their control schemes in commercial software codes used in the automotive industry. In previous works [1–3], we presented a muscle control algorithm for aHBMs based on intermittent equilibrium point control implemented in the FE software code LS-DYNA. Due to the fact that the mapping between muscle stimulations and the body motion itself is highly non-linear and complex – because of actuation and kinetics redundancy – it is necessary to optimise the parameters for the open-loop central nervous system command and the closed-loop reflex signal before running the simulation. Tuning these parameters takes a tremendous amount of time in FE models and currently represents the bottleneck of using suggested bio-inspired control approach in FE aHBMs.

We here propose a method to perform all optimisations required to determine controller and active muscles parameters on a reduced MB model. As a consequence, the control approach from the computationally efficient MB model can be transferred to the computationally demanding full FE model, provided both models have similar muscle material and controllers. This allows to reduce the simulation time needed and promises the feasibility of bio-inspired motor control models in full FE aHBMs in the near future.

2 Methods

Muscloskeletal model. To demonstrate the applicability of the proposed approach, the right upper extremity from the THUMS v.5 [4] FE aHBM was extracted, simplified and transformed into a MB model in LS-DYNA, see Fig. 1. Later it was parametrized to match a Matlab/Simulink MB model [5] which has two degrees of freedom in the sagittal plane and consists of an upper and a lower arm connected by the elbow and shoulder joint. It is actuated by six lumped muscles modelled with an extended Hill-type muscle material [1, 3], muscle activation dynamics from [2] and a controller described below.

Fig. 1: The upper extremity model from the THUMS v.5 aHBM [4]: transformation steps from full FE to MB. Full FE model with soft tissues a); FE model with deformable bones consisting of various materials b); FE model with bones made from rigid material c).
Hybrid Equilibrium Point Controller. The intermittent hybrid equilibrium point control model hypothesises that voluntary movements are generated by switching between stable equilibrium positions [6]. To this end, the stimulation \( u_i \) for each muscle \( i \) is calculated as

\[
u_i(t) = u_i^\text{open}(t) + u_i^\text{closed}(t) = u_i^\text{open}(t) + \frac{k_{p,i}}{k_{opt,i}}(l_i^\text{CE}(t) - \lambda_i) + \frac{k_{d,i}}{k_{opt,i}}v_i^\text{CE}(t - \delta_i),
\]

where \( k_{p,i} \) and \( k_{d,i} \) – feedback gains, \( l_i^\text{CE} \) and \( v_i^\text{CE} \) – length and contraction velocity of the contractile element of muscle \( i \). The open-loop stimulation \( u_i^\text{open} \) and the desired lengths and velocities of the muscle contractile elements \( (\lambda_i \text{ and } \dot{\lambda}_i) \) define the equilibrium positions and need to be determined by optimization. Applying this controller for one equilibrium point results in a stable joint configuration. By switching between equilibrium points, the biomechanical characteristics of the muscles generate a dynamic movement.

Optimisation Strategy. In a first step, joint configurations which represent an equilibrium point are derived, e.g. the initial and end configuration of a movement from the experimental data. Based on these joint angles, the open-loop stimulations for all muscles are optimized such that all net joint torques are equal to zero, meaning that the system is in equilibrium. In this way, an optimization problem is defined with a cost function minimizing the sum of all muscle stimulations and the constraint of an equilibrium [2]. By solving the problem, we can determine combinations of open-loop stimulations \( u_i^\text{open} \) which define the desired muscle lengths \( \lambda_i \) corresponding to the equilibrium point. To model point to point movements, the desired velocities were all set to zero \( \dot{\lambda}_i = 0 \). The solution was done for the MB model with non-linear-linear constraint algorithms available in Matlab. Once all control parameters were determined, they were then transferred to the FE model to obtain the final solution.

3 Results and Discussion

Two calibration runs were performed with fully stimulated extensors for one and flexors for another to determine the actual range of motion of the MB arm model. After that, three target points lying on a same horizontal line were determined for the left, right and centre points, and optimization was performed to find appropriate stimulation levels for the muscles to reach desired positions. As expected, during optimization for the MB model in LS-DYNA it took approximately 6 seconds of calculation time per one iteration of the 3 seconds of real time movement on 2 cores of the Intel(R) Xeon(R) CPU E5-1620 @ 3.60GHz. Retrieved stimulation levels were applied to the active muscle elements in the FE model with soft tissues leading to the results shown in Fig. 2. Calculation time for the before-mentioned model was 3 days on 16 cores of the AMD(R) Opteron(R) CPU 6320 @ 3.2GHz, but with share resources among users.

After analysis of the results, we can conclude that the implementation of the MB model in LS-DYNA was successful leading to expected stable equilibrium point positions, but with some remarks. First, the targeted end positions for MB and FE models are different due to the difference in joint modelling. Thus, the MB should be updated accordingly. Second, for the initial 1.25 seconds of simulation with the FE model transient process takes place which occur due to incorrect stiffness of the soft tissues in the model taken from cadaver experiments. Consequently, the FE model materials should be adjusted to represent tissue of a living human. Later studies would include an introduction of more muscles and a simulation of 3D movements.

Acknowledgements This work was supported by the German Research Foundation (DFG) within the Cluster of Excellence “Data-Integrated Simulation Science” (EXC 2075) at the University of Stuttgart and by the Ministry of Science, Research and the Arts Baden-Wuerttemberg through project Az:33-7533.-30-20/7/2.

References

[1] D. F. B. Haeufle, M. Günther, A. Bayer, and S. Schmitt, Journal of Biomechanics 47(Apr), 1531–1536 (2014).
[2] A. Bayer, S. Schmitt, M. Günther, and D. F. B. Haeufle, Computer Methods in Biomechanics and Biomedical Engineering 20(8), 803–821 (2017), PMID: 28387534.
[3] C. Kleinbach, O. Martynenko, J. Promies, D. F. B. Haeufle, J. Fehr, and S. Schmitt, BioMedical Engineering OnLine 16(1) (2017).
[4] M. Iwamoto, Y. Nakahira, and H. Kimpara, Traffic injury prevention 16(sup1), S36–S48 (2015).
[5] D. Driess, H. Zimmermann, S. Wolfen, D. Suissa, D. F. Haeufle, D. Hennes, M. Toussaint, and S. Schmitt, Learning to Control Redundant Musculoskeletal Systems with Neural Networks and SQP: Exploiting Muscle Properties, in: 2018 IEEE International Conference on Robotics and Automation (ICRA), (IEEE, Brisbane, may 2018), pp. 6461–6468.
[6] D. A. Kistemaker, A. J. Van Soest, and M. F. Bobbert, Journal of Neurophysiology 95(May), 2898–2912 (2006).