Available online at www.sciencedirect.com

ScienceDirect

IFAC-PapersOnLine 49-32 (2016) 200–203

EMG-driven models of human-machine interaction in individuals wearing the H2 exoskeleton ⋆

Guillaume Durandau * **Massimo Sartori** * **Magdo Bortole** ** **Juan C. Moreno** ** **José L. Pons** **,*** **Dario Farina** ****

* *Institute of Neurorehabilitation Systems, Universitätmedizin Göttingen, Göttingen, Germany (e-mail: guillaume.durandau@bccn.uni-goettingen.de).*
** *Cajal Institute, CSIC, Madrid, Spain*
*** *Tecnológico de Monterrey, Monterrey, Mexico*
**** *Department of Biomedical Engineering, Imperial College London, UK*

Abstract: EMG-driven modeling has been mostly used offline and on powerful desktop computers, limiting the application of this technique to neurorehabilitation settings. In this paper, we demonstrate the use of EMG-driven modeling in online (i.e. in real-time) running on a fully portable embedded system and interfaced concurrently with a powered lower limb exoskeleton. This work provides evidence of the feasibility of real-time model-based control of complex multi-joint exoskeleton system, thus opening new avenues for personalised robot-aided rehabilitation interventions.

Keywords: Exoskeleton, EMG, modeling, wearable, human-machine interface

1. INTRODUCTION

Electromyography (EMG-)driven modeling has shown good results in predicting internal body forces including joint torquesSartori et al. (2011), joint compressive forcesGerus et al. (2013), joint stiffnessSartori et al. (2015) or muscle forces. Accessing these variables experimentally is difficult and/or needs invasive methods. Nevertheless, real-time estimates of internal biomechanical variables would enhance an individual's rehabilitation process and maximize treatment outcome. Furthermore, accessing internal mechanical variables would enable interfacing an individual's musculoskeletal system with wearable devices such as powered exoskeletons (see Bortole et al. (2015)) that could support the rehabilitation process. In this context, therapeutic robotic devices are expected to grow into 1.7 billion market by 2025, as an aging population demands better quality of life and superior performance (according to Lux Research [1]). Hence, wearable robotic technologies for humans are called to become ubiquitous in the next 20-25 years. Available prototypes and marketed devices still require ability to perceive and detect the actual users effort. Current solutions achieve this indirectly by detecting external forces that are produced once movement and interaction has occurred. A solution for detection of the human subjects motor intention and force production would represent a change of paradigm in human-robot interaction.

On the other hand, the behaviour and the response of therapeutic exoskeletons to the user's actions ought to be adaptable rather than pre-programmed, to generate the required assistance as a function of both the capability of the user and the rehabilitation outcome. One well accepted strategy to promote neural plasticity in exoskeleton-based robot therapy is the impedance-based assistance. It relies on restoring forces, which are generated using the concept of mechanical impedance when the users deviate from the desired trajectory. In such type of robots, feedback loop of interaction forces can be implemented to achieve relatively slow assist-as-needed coordination under an adaptive control structure. It can be hypothesized that assist-as-needed control strategies in wearable exoskeletons informed by EMG-based estimates of joint torque can represent improvements in system function to a) adequately adapt to changing user and task condition and b) improving robotic interventions by enhancing motor adaptation (in case of patient users).

With these ideas in mind we created a real-time EMG-driven modeling algorithm, that we implemented on a low power, battery powered wearable system (Raspberry Pi Foundation, UK). We then interfaced it in real-time to the H2 lower-limb exoskeleton (Technaid S.L., Spain). We here provide system design and performance results.

2. METHOD

2.1 Real-Time modeling

We extended the offline Calibrated EMG-informed Neuromusculoskeletal Modeling (CEINMS) toolbox (see Piz-

⋆ This work was supported by the ERC Advanced Grant DEMOVE [267888]
[1] `http://mybs.in/2TDiWNJ`

2405-8963 © 2016, IFAC (International Federation of Automatic Control) Hosting by Elsevier Ltd. All rights reserved.
Peer review under responsibility of International Federation of Automatic Control.
10.1016/j.ifacol.2016.12.214

zolato et al. (2015)) for creating our real-time EMG-driven model. We included direct tcp/ip connection to an EMG amplifier (OTBioelettronica, Italy) for having access in real-time to EMG data. Experimentally recorded joint angles (See Section 3) were input into a Multidimensional Cubic B-Spline (MCBS) software (see Sartori et al. (2012)). MCBS computes the muscle-tendon length (Lmt) and moment arms (MA) from the joint angles. Using muscle kinematics information and EMG, CEINMS compute muscle-tendon force and joint torque. Furthermore, a calibration procedure is used to identify model parameters that vary non-linearly across individuals' anthropometrics. The calibration is an optimization procedure that minimize the torque error between experimental and predicted joint torques. The calibration software also computes the spline coefficients for MCBS. In this context, experimental joint angles and torque data are recorded using the exoskeleton sensors as described below.

2.2 H2 exoskeleton

The H2 is a lower limb exoskeleton possessing six degrees of freedom (DOFs) including left/right hip flexion-extension, knee flexion-extension, and ankle plantar-dorsi flexion. Each degree of freedom of the H2 has position sensor, interaction torque sensor and motor torque sensor. These informations are send to our wearable system through controller area network (CAN) bus. The joints are actuated by an electric motor and an harmonic drive that can provide a continuous net torque up to 35 Nm or a peak torque of 180 Nm for short periods of time (< 1 second). For providing feedback to the practitioner, a graphical user interface (GUI) was implemented using tcp/ip to connect to the wearable system via wifi. This GUI provide feedback on the EMG-driven torque, interaction torque, motor torque and the joint angle via a 3D model of the skeletal system of the patient (see figure 2).

3. EXPERIMENTS

EMGs were recorded at 2048Hz using a 256-channel EMG amplifier (OTBioelettronica, Italy) from 4 muscle groups: gastrocnemious medialis and lateralis, soleus and tibialis anterior. The joint angles and joint torques were recorded using the H2 exoskeleton position sensors at 100 Hz. The EMG-driven musculoskeletal model we used has 3 DOFs (right knee, right ankle and right subtalar) and 13 muscles. In this experiment all the joints were computed but only the ankle muscles received EMGs signal. The wearable system used was a Raspberry pi 2 with a four-core processor running at 900Mhz and 1GB of RAM. The EMG amplifier, practitioner desktop and wearable system were connected by WiFi. We first scaled the OpenSim model using motion capture data (Qualysis, Sweden) recording at 128Hz for the marker and two in-ground force plates (Bertec, USA) recording at 2048Hz. The scaled OpenSim model was needed for the computation of the MCBS spline coefficients (see Section 2.1). We then calibrated the EMG-driven model using the EMG, MA, Lmt, and torque from the interaction torque sensors recorded by the H2. During the recording the joint were blocked in position thus the subject produces only isometric force. The real-time experiments were done as follow:

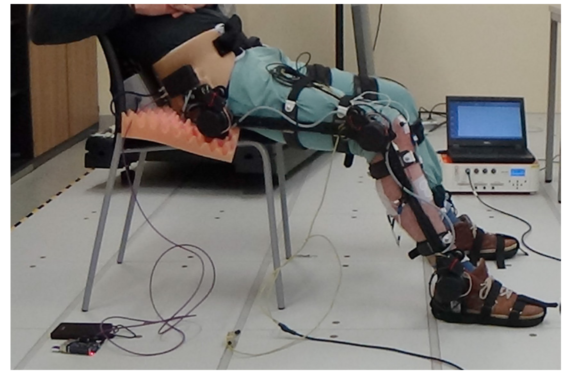


Fig. 1. Setup of the experiment. The raspberry pi 2 is on the floor with the battery connected to it. The raspberry pi 2 is connected to the H2 via CAN bus. The EMG amplifier is in the background and the user with the exoskeleton is on a chair.

- The first experiment was to do isometric movements against the exoskeleton joint that were blocked in fixed position.
- In the second experiment, the subject was requested to move the ankle joint with the zero impedance controller of the H2 exoskeleton.

4. RESULT

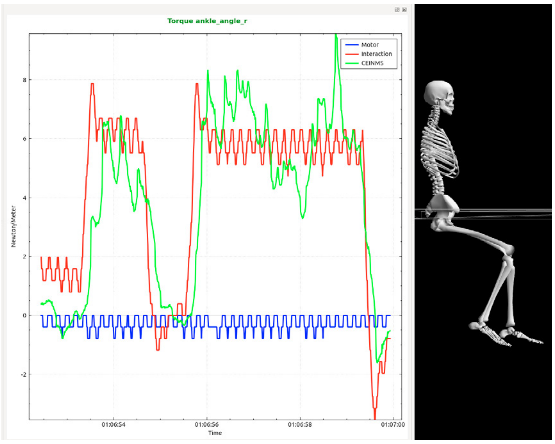
The first experiment was primarily done to verify the success of the calibration procedure on the same condition namely isometrics condition. The second experiments was done to see if the calibration procedure which was done on isometrics condition can extrapolate to dynamics condition.

The figure 3a shows the histogram of the spline computation time on a desktop computer in seconds (the Y axis is the normalisation of the total number of samples). The mean computational time is 0.36 ms for this trial. The figure 3c shows the same result but on the wearable system. The mean computational time is 4.26 ms for this trial. The figure 3b shows the histogram of the EMG-driven model computation on a desktop computer in seconds. The mean computational time is 0.24 ms for this trial.The figure 3d shows the same result but on the wearable system. The mean computational time is 2.63 ms for this trial.

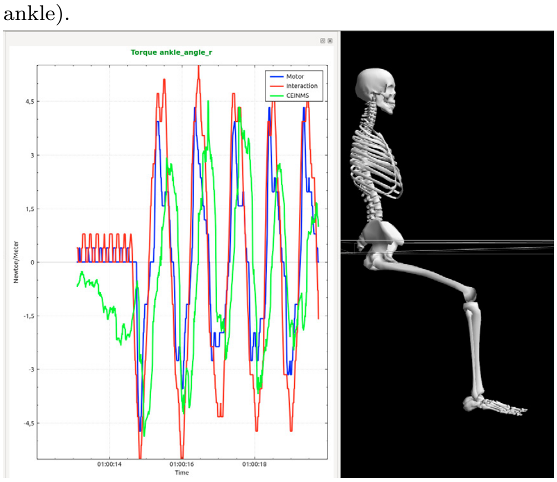
Figure 2 shows the result of the real-time EMG-driven model and the GUI on the practitioner side for the ankle joint. Figure 2a is the resulting torque with the ankle joint blocked by a position controller (0 degree). Figure 2b is the resulting torque with the ankle joint driven by a zero impedance controller. The EMG predicted torque accuses a delay of around 150-200 ms from the torque reported by the interaction torque sensor.

5. DISCUSSION

The figure 3 shows that the computational time of the real-time EMG-driven model on a wearable system is $\approx$ 10 times slower compared to a desktop computer. Furthermore, with a mean of 6.89 ms for the computation time of the spline and the EMG-driven model, we are under the electromechanical delay and thus still being in



(a) Result of the EMG-driven model with the joint of the ankle blocked in position (the user cannot move the ankle).

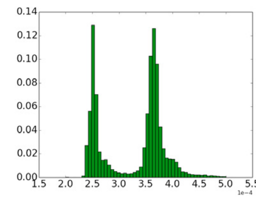


(b) Result of the EMG-driven model with the a zero impedance control (the user can freely move the exoskeleton).

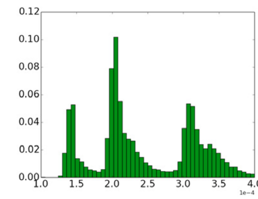
Fig. 2. Result of the EMG-driven model displayed on the GUI, with the blue line representing the motor torque, the red line representing the interaction torque and the green line representing the EMG-driven torque. The Y axis is in Newton/meter and the X axis in second. The EMG-driven torque have a delay of around 150-200 ms.

the prediction zone. This is an important point because with this algorithm we can be informed of the DOFs torque of the patient before actual movement execution. With the torque and internal body parameters information in advance, a rehabilitation controller can compute the best helps depending of the specific weakness of the patient. Furthermore, a controller with no delay will give the exoskeleton a more natural and pleasant experience to the user.

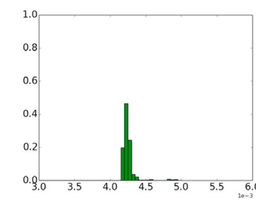
In Figure 2 we can see that the amplitude of the interaction torque is followed by the EMG-driven model. We can also see a small delay of around 150-200 ms which is due to the fact that the EMG amplifier sent us packet of data by tcp/ip connection over wifi which add a delay that can not be reduce as long as this EMG amplifier system is use.



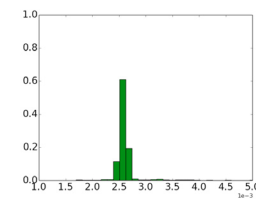
(a) Spline computation time on desktop (in seconds with a $1e^{-4}$ factor).



(b) Modeling computational time on desktop (in seconds with a $1e^{-4}$ factor).



(c) Spline computation time on Raspberry pi 2 (in seconds with a $1e^{-3}$ factor).



(d) Modeling computational time on Raspberry pi 2 (in seconds with a $1e^{-3}$ factor).

Fig. 3. Computational time from the real-time algorithm on desktop computer and on wearable system.

6. CONCLUSION

Further work is needed to fully integrate a real-time, wearable EMG amplifier. The EMG amplifier used in our experiments is limiting the full usage of the wearable system by not being wearable and not real-time. More work need also to be done to close the loop with the exoskeleton. In this abstract we show the open-loop system, namely we do not not use the information computed to control the torque outputted by the motor. Closing the loop by controlling the motor torque using the information computed by the EMG-driven model can create a new generation of controller that know in advance the torque that will be produce by the joint of the patient before movement execution takes place thanks to the electromechanical delay.

Finally, more computed variables can be added to the model like the joint compression forces (see Gerus et al. (2013)), joint stiffness (see Sartori et al. (2015)). The joint compression forces can be useful for exoskeleton as a variables that need to be minimized because a too much high compression force is a sign of unhealthiness of the joint that can lead to precocious joint deterioration. Joint stiffness is also a variables that can be use as an assessment by the exoskeleton of the fact that the muscles do not produce enough force or too much co-contraction that create stiff or wobbly gait pattern. Furthermore, muscle primitives can be use to drive the EMG-driven model (see Sartori et al. (2013)). Primitives can be helpful to reduce the numbers of electrodes needed to compute the joint torque. Also it can help in the in conditions of weak or abnormal muscle activation, such as incomplete spinal cord injuries or paresis after brain damage.

REFERENCES

Bortole, M., Venkatakrishnan, A., Zhu, F., Moreno, J.C., Francisco, G.E., Pons, J.L., and Contreras-Vidal, J.L. (2015). The h2 robotic exoskeleton for gait rehabilitation after stroke: early findings from a clinical study.

Journal of neuroengineering and rehabilitation, 12(1), 54.

Gerus, P., Sartori, M., Besier, T.F., Fregly, B.J., Delp, S.L., Banks, S.A., Pandy, M.G., D'Lima, D.D., and Lloyd, D.G. (2013). Subject-specific knee joint geometry improves predictions of medial tibiofemoral contact forces. *Journal of biomechanics*, 46(16), 2778–2786.

Pizzolato, C., Lloyd, D.G., Sartori, M., Ceseracciu, E., Besier, T.F., Fregly, B.J., and Reggiani, M. (2015). Ceinms: A toolbox to investigate the influence of different neural control solutions on the prediction of muscle excitation and joint moments during dynamic motor tasks. *Journal of biomechanics*, 48(14), 3929–3936.

Sartori, M., Gizzi, L., Lloyd, D.G., and Farina, D. (2013). A musculoskeletal model of human locomotion driven by a low dimensional set of impulsive excitation primitives. *Frontiers in computational neuroscience*, 7.

Sartori, M., Maculan, M., Pizzolato, C., Reggiani, M., and Farina, D. (2015). Modeling and simulating the neuromuscular mechanisms regulating ankle and knee joint stiffness during human locomotion. *Journal of Neurophysiology*, 114(4), 2509–2527.

Sartori, M., Reggiani, M., Farina, D., and Lloyd, D.G. (2011). Emg-driven forward-dynamic estimation of muscle force and joint moment about multiple degrees of freedom in the human lower extremity. *PloS one*, 7(12).

Sartori, M., Reggiani, M., van den Bogert, A.J., and Lloyd, D.G. (2012). Estimation of musculotendon kinematics in large musculoskeletal models using multidimensional b-splines. *Journal of biomechanics*, 45(3), 595–601.