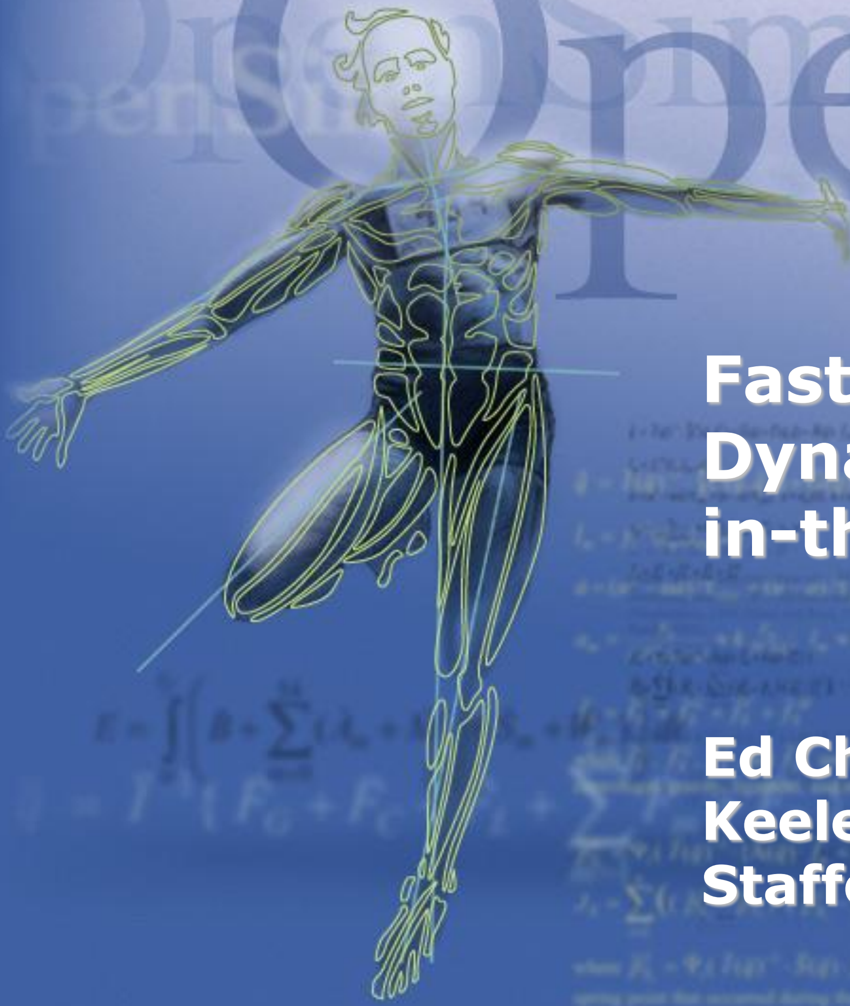


Fast Simulation of Arm Dynamics for Real-time, User-in-the-loop Control Applications

Ed Chadwick
Keele University
Staffordshire, UK.

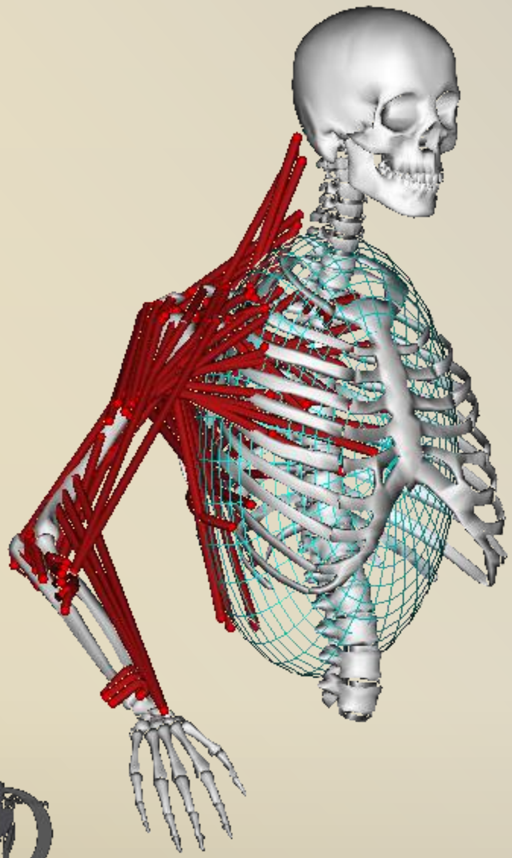
Acknowledgements

- Dimitra Blana, Keele University, Staffordshire, UK.
- Ton van den Bogert, Cleveland State University, Cleveland, Ohio.
- Robert Kirsch, Case Western Reserve University, Cleveland, Ohio.

- OpenSim project leads, team members and developer community
- NCSRR Outstanding Researcher Award scheme

- National Institutes of Health

Fast Simulation of Arm Dynamics for Real-time, User-in-the-loop Control Applications



Overview

1. Motivation and background work
2. Methods used to build the model and achieve fast simulation
3. Examples of real-time simulations and user-in-the-loop experiments
4. Discussion of achievements and limitations of our approach

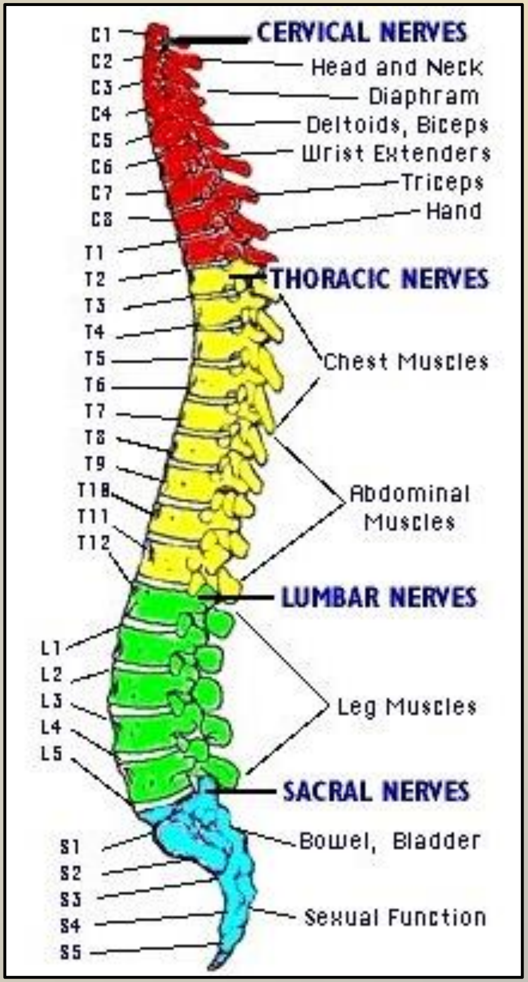
1. Motivation and background work

Restoration of function in spinal cord injury

- The long-term goal of this work is to restore natural control of arm movement to people with high-level spinal cord injury (or other neuromuscular disorders that cause paralysis).
- There are many approaches to this:
  - Robotic exoskeletons
  - Robotic assistive devices
  - Regenerative medicine
  - Re-animation using muscle stimulation
- Recently identified as a grand challenge by Nesta
  - Longitude Prize: http://www.longitudeprize.org/challenge/paralysis

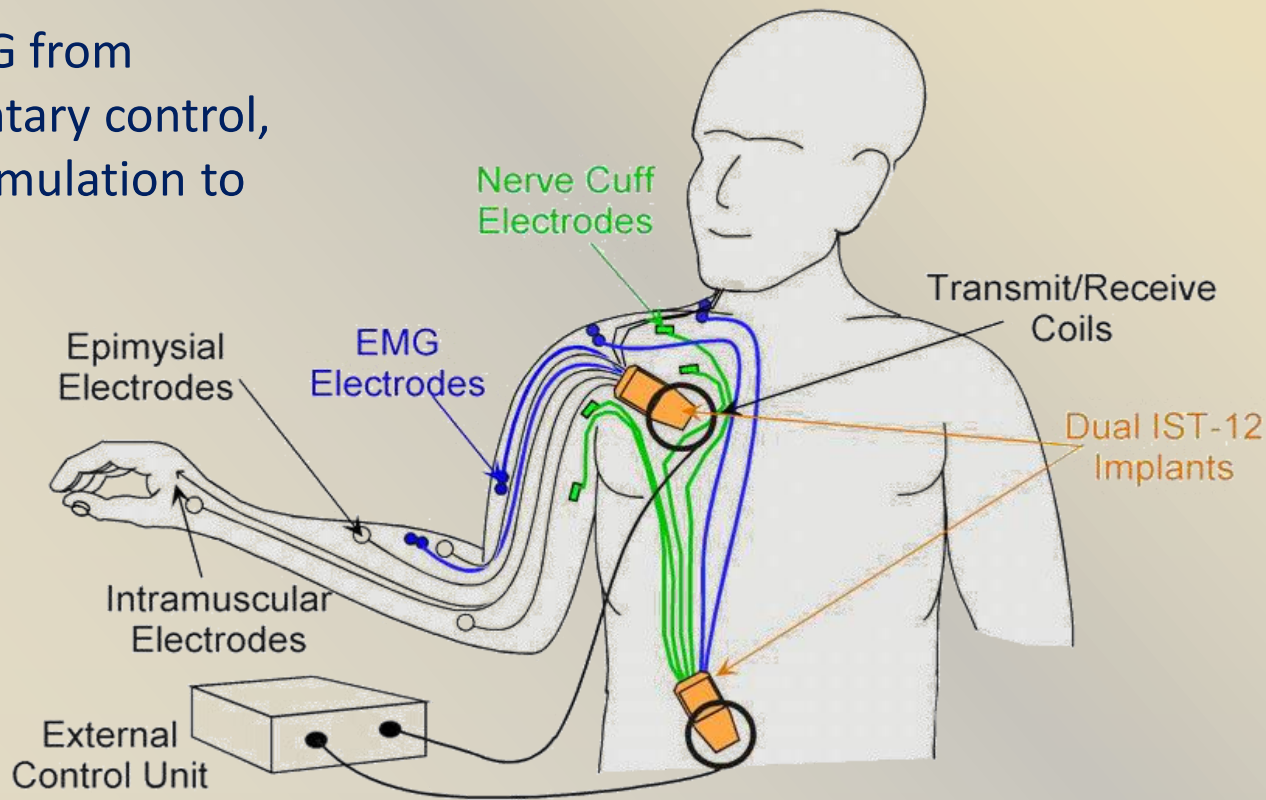
Paralysis after spinal cord injury

- Spinal cord injury causes an interruption to the command signals from the brain reaching the muscles that control movement
- The muscles themselves, however, are still capable of producing force
- Coordinated stimulation of the muscles and nerves can produce functional movements in people who have lost voluntary control of movement



An implanted neuroprosthesis system

In this example, EMG from muscles under voluntary control, is used to control stimulation to paralysed muscles.



Case Western Reserve University / Cleveland FES Center

An external controller determines the required muscle activation patterns to achieve the desired movements.

Challenges in the control of assistive devices

- Each person's requirements and limitations are unique. How do we design a device to accommodate that?
- How does a user control an assistive device?
- How does this work in people with very high level injuries, whose command sources may be limited?
- How can we ensure that the device will operate safely?
- How will the system respond if the user's characteristics (e.g. strength) change over time?

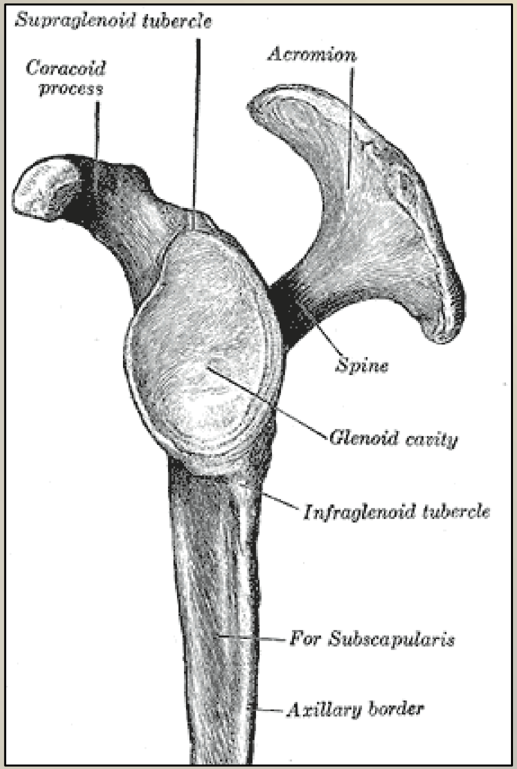
We use model-based design to address these challenges.

Natural control of assistive devices

- Command interfaces
  - Sip-and-puff (mouth controlled devices)
  - Head orientation
  - Eye tracking
  - EMG from voluntary muscles
  - Brain-computer interfaces
- Feedforward and feedback control
  - Feedforward control identifies the ideal pattern of muscle activations required to achieve a desired movement
  - Feedback control corrects for perturbation, errors, fatigue during the performance of the movement

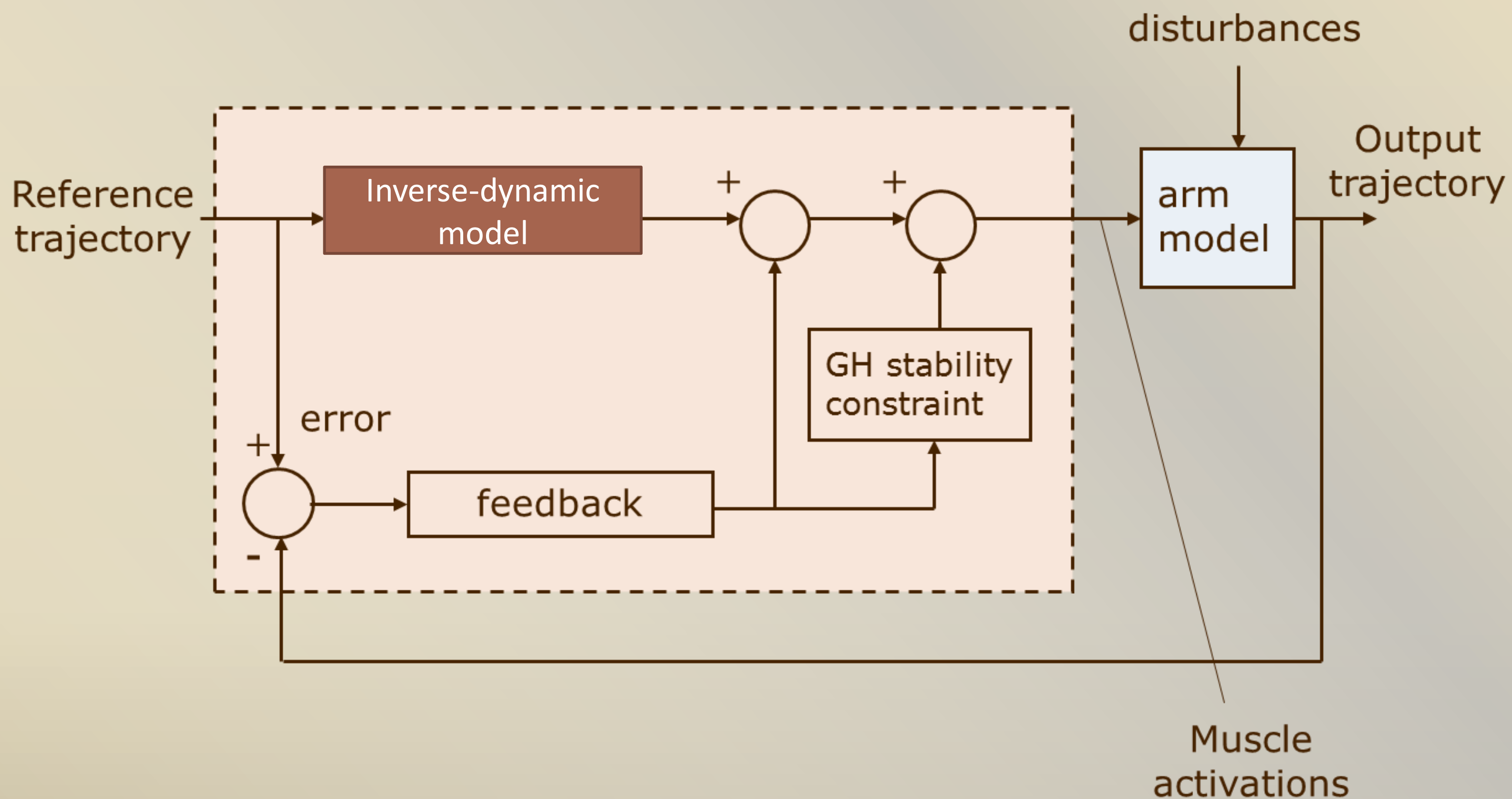
Controlling the shoulder girdle

- We need to ensure that scapular motion is controlled to provide a stable base for elevation of the arm and hand positioning
- We need to ensure that glenohumeral stability is maintained
- Both of these are the responsibility of the controller in an FES system



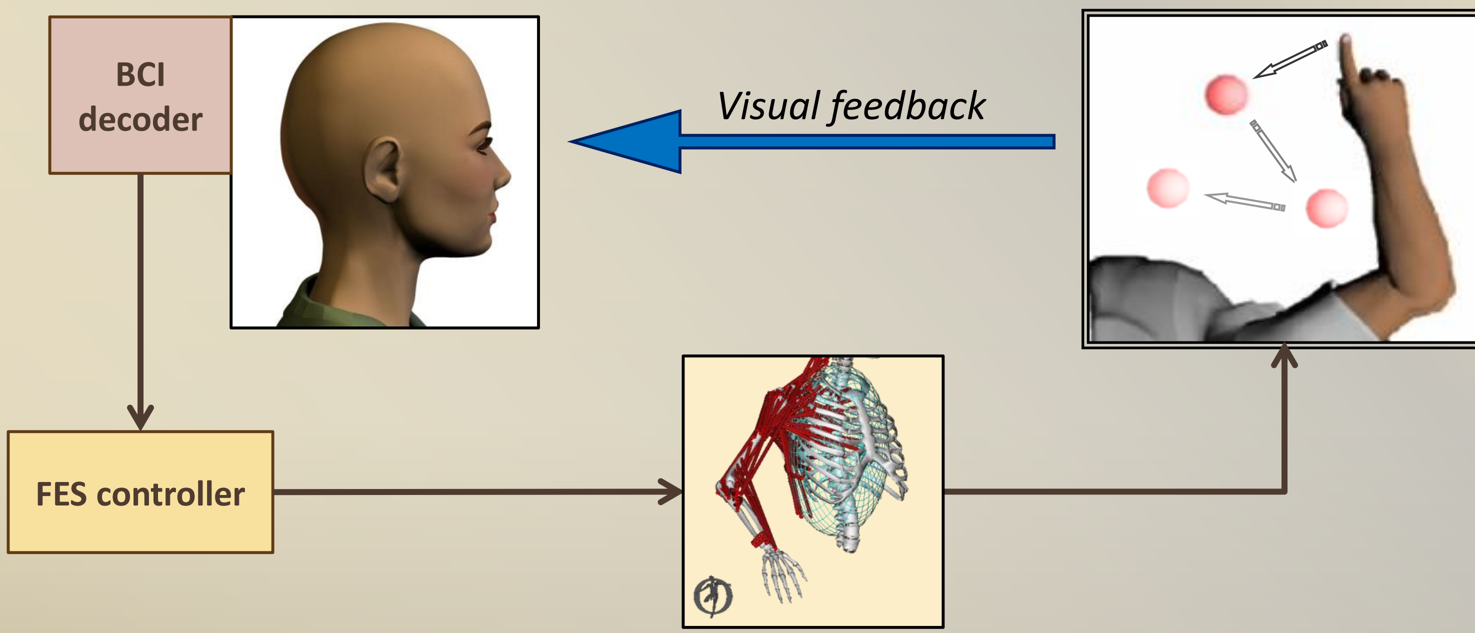
Musculoskeletal models in device development

- We have used musculoskeletal models (offline) in controller design



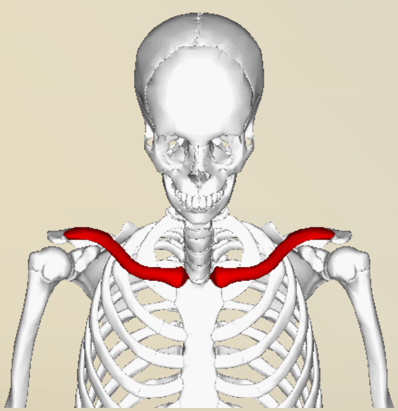
The user as an integral part of the control loop

- For high-level injuries, a brain-computer interface becomes a promising command source, and the user becomes an integral part of the control loop
- A forward-dynamic model of the arm can be used in place of the user’s own arm for device development and testing



Stiff problems in musculoskeletal simulations

- Large variation in time-constants within the system (rate of response of outputs for specified changes in inputs) leads to stiff systems
  - integrator step sizes have to be very small to ensure a stable simulation
  - small step sizes lead to many steps and therefore slow simulations



Low-inertia clavicle controlled by stiff muscles

The aims of this work were to develop a comprehensive model of the shoulder and upper limb representing muscle dynamics, muscle-skeleton coupling and arm inertial properties that runs in real time.

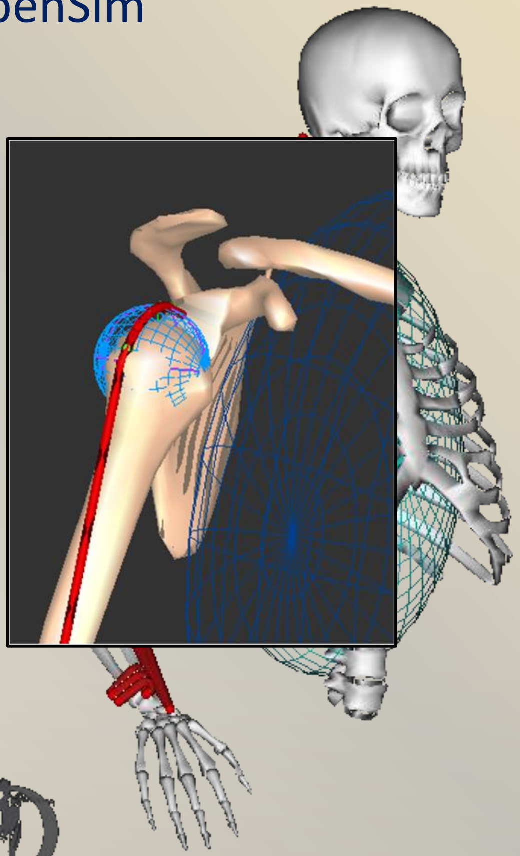
2. Building and simulating the model

Building the model in OpenSim

- Model was originally built in SIMM, based on Delft Shoulder and Elbow Model (van der Helm, 1994), then converted to OpenSim

- 11 Degrees of Freedom
- 31 muscles, 138 muscle elements
- Wrapping objects defined around bones

- Hill-type muscle model
- First-order muscle dynamics



Ensuring fast simulation of complex structures

- Use an implicit solver to address time-step problem
  - Allows much bigger integration steps to be taken for a stiff system

$$x_{n+1} = \Delta t \, f(t_{n+1}, x_{n+1}) + x_n$$

- Use analytical derivatives of state variables
  - Computationally faster than numerically estimating derivatives

- Pre-process moment arms and muscle lengths throughout workspace
  - Run-time calculation of muscle wrapping can be a time-consuming process

Pre-calculation of muscle lines-of-action

- Muscle moment arms and lengths pre-calculated and exported using OpenSim API, while moving the model through its entire workspace

```
for istep = 1:size(angles,1)
    for idof = 1:nDofs
        currentDof = CoordSet.get(idof-1);
        currentDof.setValue(state,angles(istep,idof),1);
    end

    for imus = 1:nMus
        length(istep,imus) = MuscleSet.get(imus-1).getLength(state);
    end
end
```

- Properties modelled using non-linear (max 4^{th} order) polynomials with errors less than 10% of maximum values, or 2mm (whichever is greater).

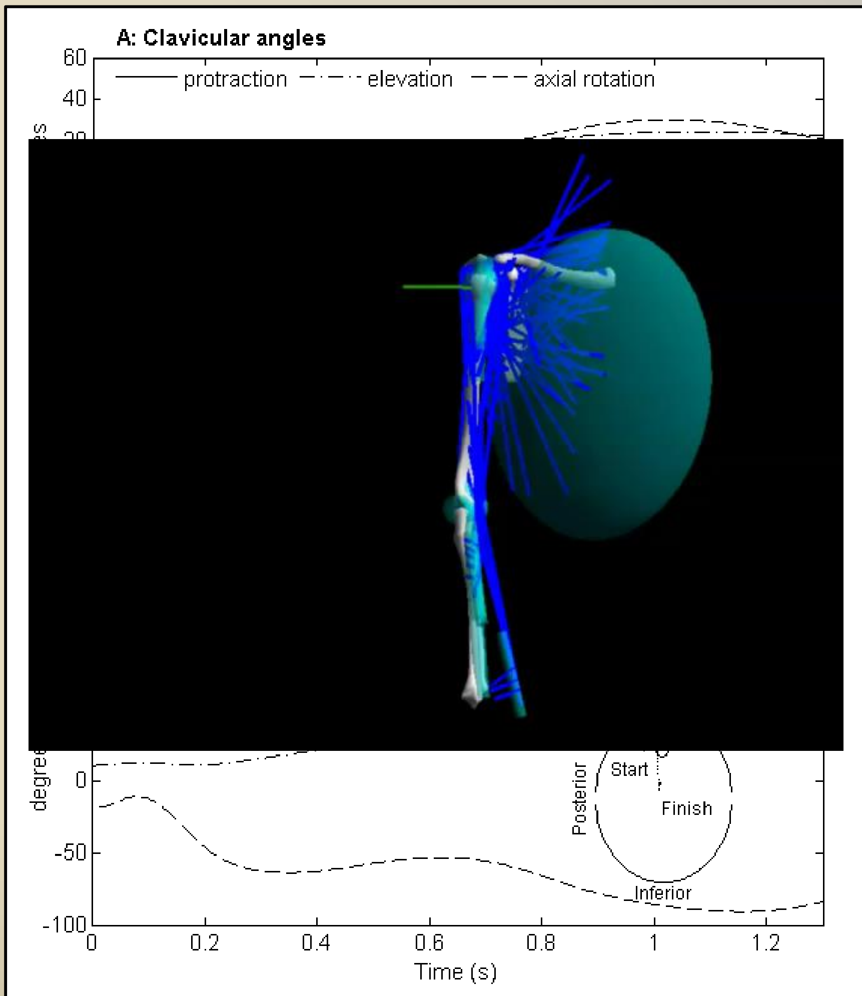
Implementation of real-time method

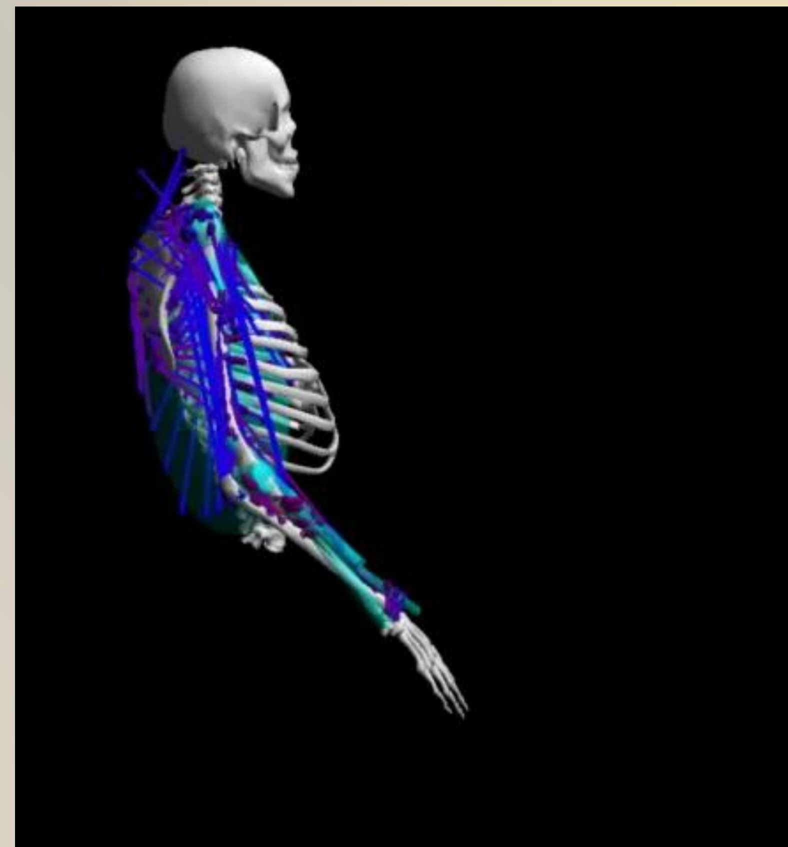
- Equations of motion and (analytical) derivatives for the model were calculated using Autolev (Online Dynamics Inc., Sunnyvale, CA)
- Muscle dynamics and derivatives were implemented using custom C-code
- Scapulo-thoracic contact was modelled using a non-linear elastic force
- Simulation was carried out in Matlab using an implicit solver (first-order Rosenbrock method)
  - Implicit method allows us to take much larger steps than explicit
  - Use of analytical derivatives speeds up step calculation

Pre-processing in OpenSim → Real-time simulation in Matlab → Visualisation in OpenSim

3. Results & example simulations

Measuring model performance

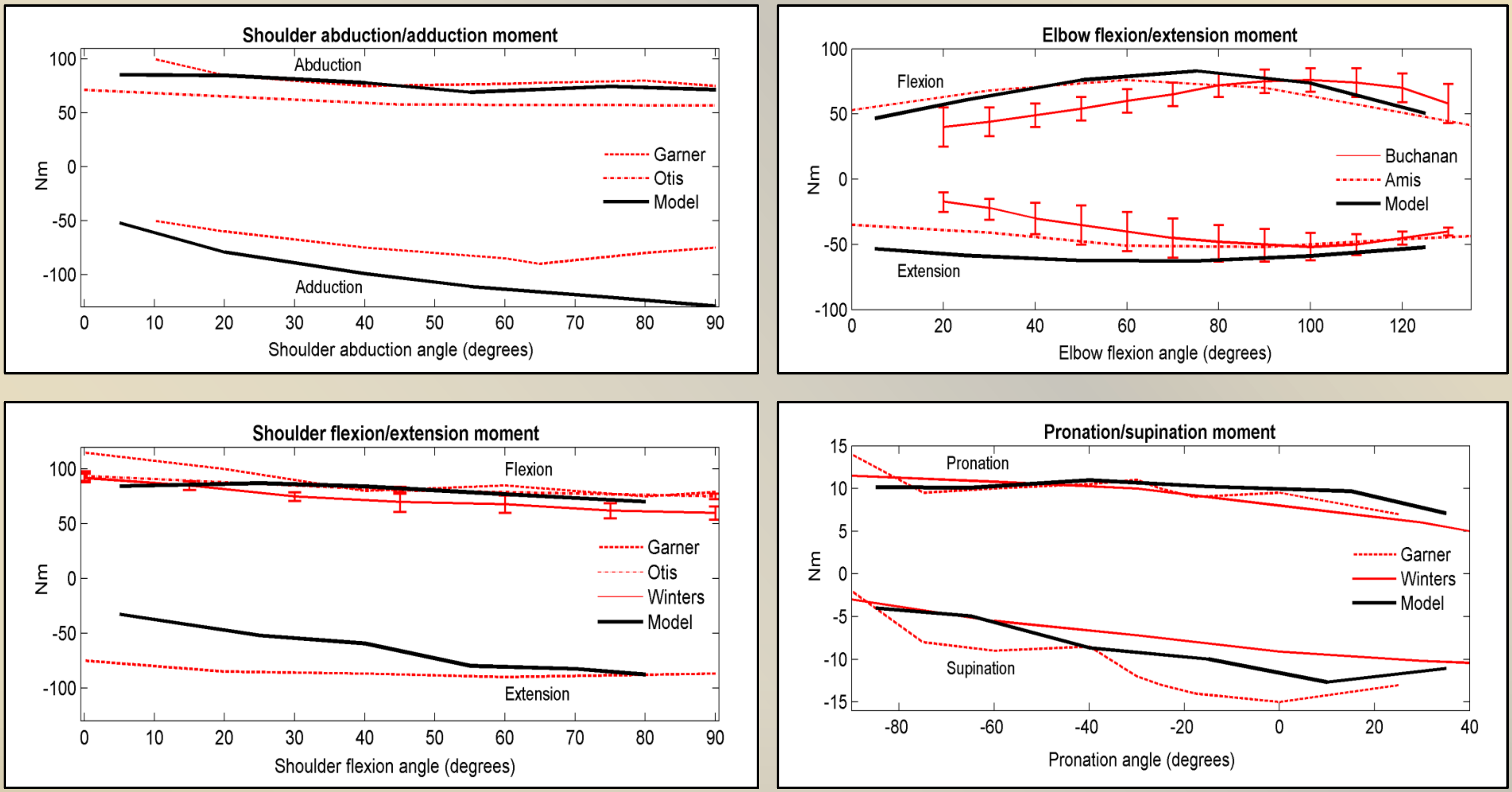




4s movement simulated in 3.5s

Forward flexion of the arm using muscle activations from inverse dynamics

Validating model behaviour



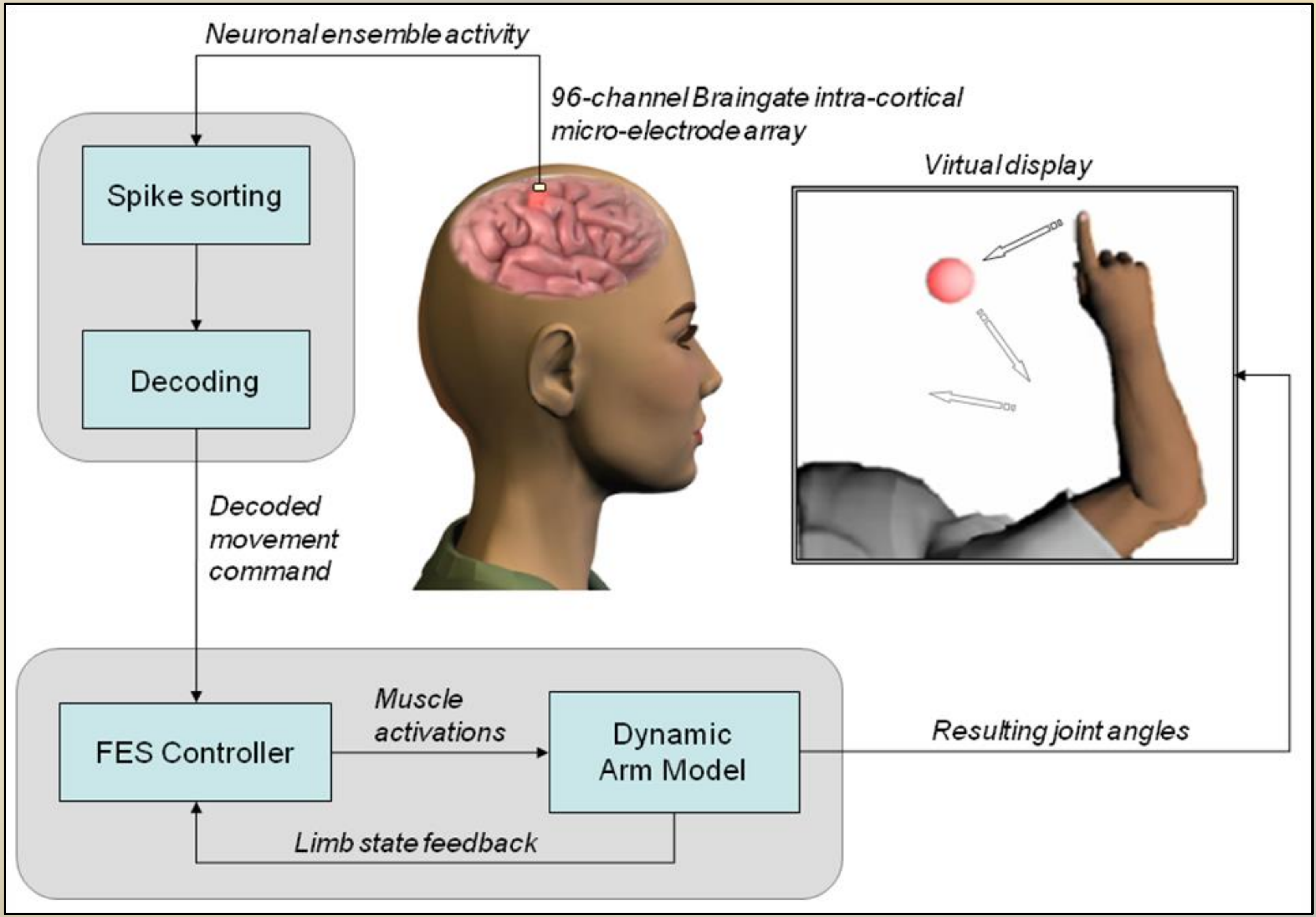
Isometric moments were maximised about each DOF and compared with literature

User-in-the-loop simulations (DAS1)

Use of real-time model in a virtual reality environment

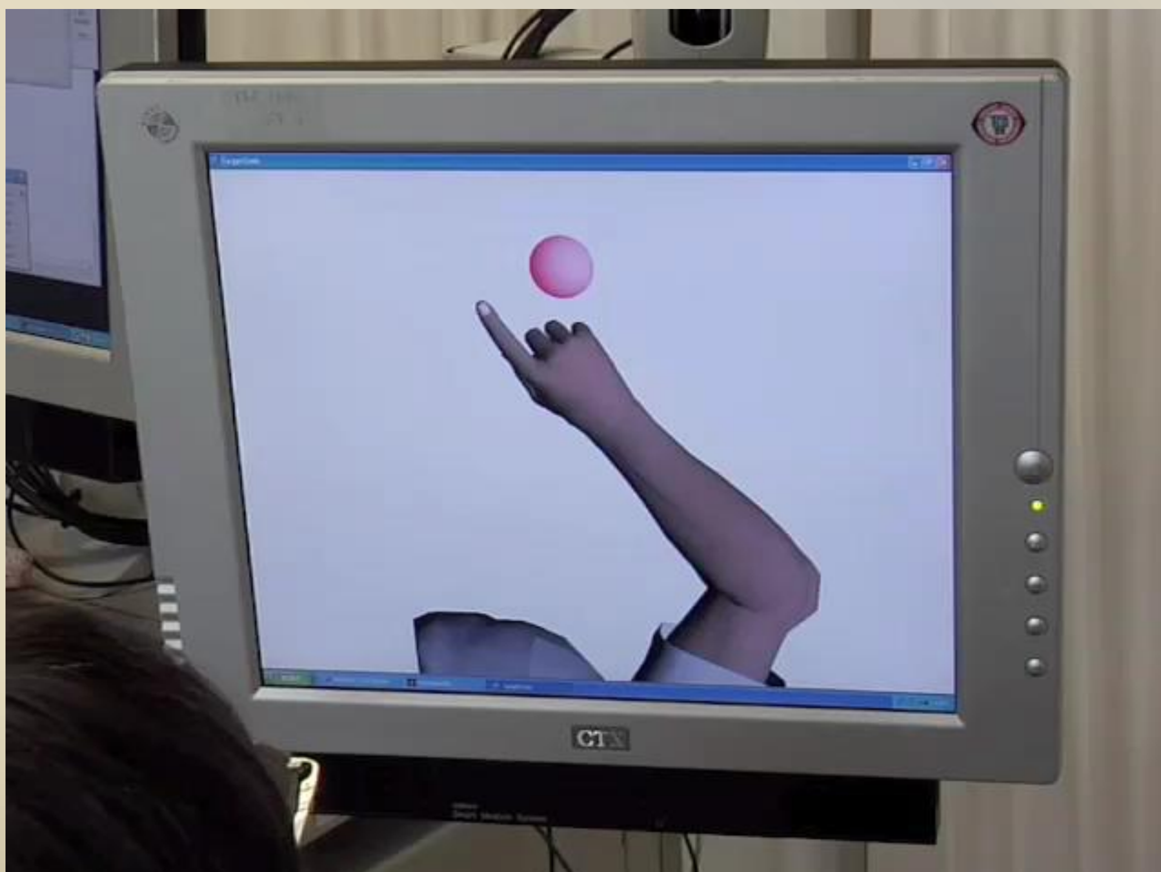
- Participant with >10 yr brainstem stroke & locked-in syndrome, user of Braingate brain-computer interface (Brown University)
- Established BCI control of kinematic systems such as computer mouse
- Simplified model of dynamic arm simulator (planar movement)

How well can the user control a dynamic, non-linear system representing arm dynamics?


Neuronal ensemble activity
96-channel Braingate intra-cortical micro-electrode array
Virtual display
Spike sorting
Decoding
Decoded movement command
FES Controller
Muscle activations
Dynamic Arm Model
Resulting joint angles
Limb state feedback

Case study: BCI development

Subject with brainstem stroke was able to achieve good control of virtual arm movement using a cortical BCI



- Control of a dynamic, non-linear system
- Training by thinking about arm movement
- Control following >10yrs arm non-use

4. Discussion of results and limitations

Simulations are faster than real time

- Model approximates the dynamic behaviour of a real human arm
- Better than real time performance on ‘normal’ computer hardware
- Some additional time for calculation of additional output parameters
- Possibility to monitor GH stability in real time during the simulation

- When integrated with VR, gives sophisticated platform for virtual device development that allows
  - Investigation of participant potential and learning
  - Optimisation of location and number of stimulation channels
  - Development of controllers; testing command sources

Limitations of the current model

- Some lack of agreement in moment-angle curves
  - Model is based on cadaver data; not matched to the individual participant
  - Approximation of moment arms
- Lack of neurological components in the model
- Missing hand!
- Difficulty of validation common to all musculoskeletal models

Next steps

- Tighter integration with OpenSim
  - model building to allow easier model customisation and conversion to real time
  - improved visualisation of results for easier interpretation for both offline and user-in-the-loop simulations
- Extension of model to include neuro-muscular components and therefore ability to model wider range of pathologies
- Addition of capability to interact with the environment
  - Add hand and contact model

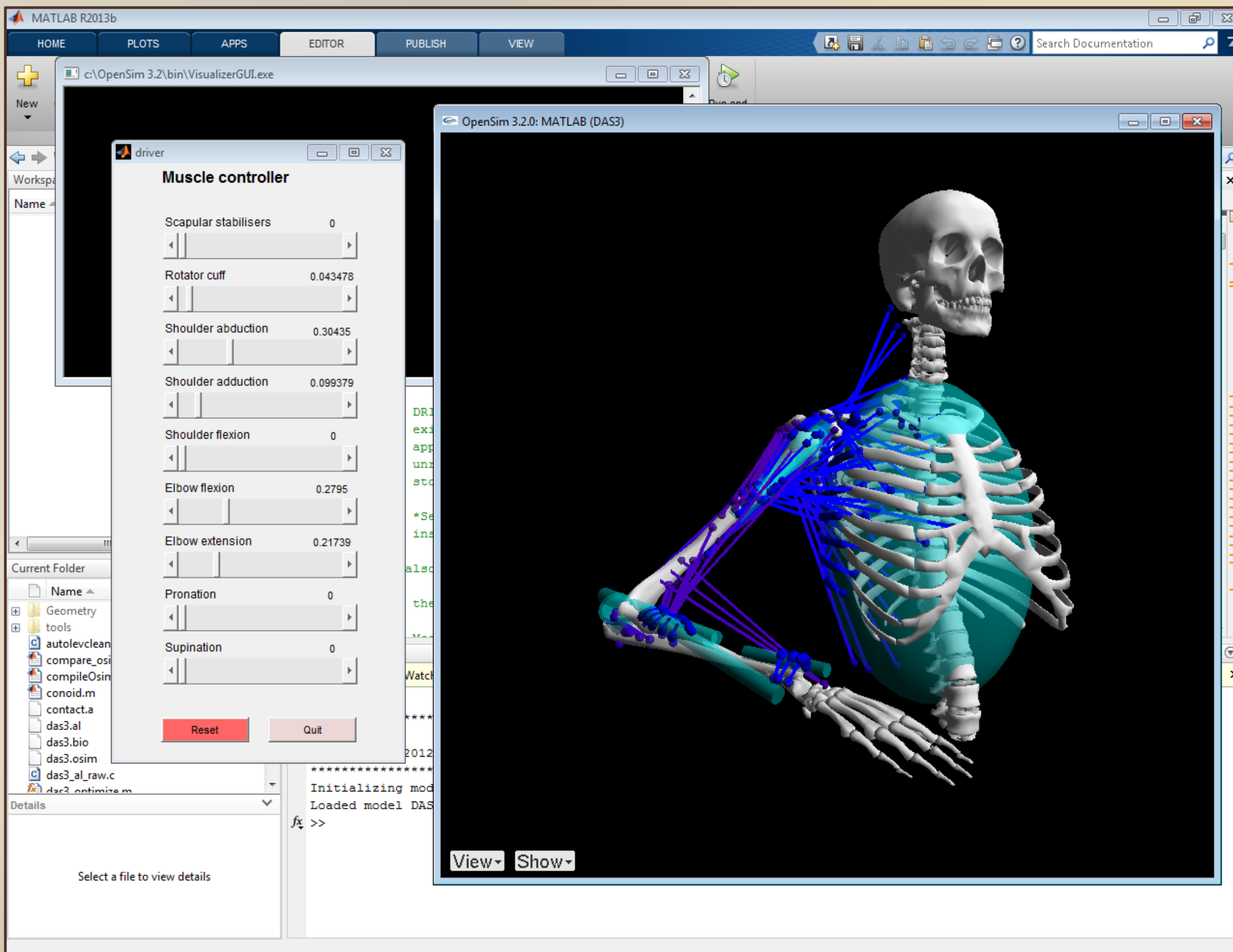
Publications

- Chadwick, Blana, Kirsch & van den Bogert (2014) Real-Time Simulation of Three-Dimensional Shoulder Girdle and Arm Dynamics. *IEEE TBME*, *In press*. http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=6755458
- Chadwick et al. (2011) Continuous neuronal ensemble control of simulated arm reaching by a human with tetraplegia. *Journal of Neural Engineering*, *8*(3), 034003. doi: http://dx.doi.org/10.1088/1741-2560/8/3/034003
- Van den Bogert, Blana, & Heinrich (2011) Implicit methods for efficient musculoskeletal simulation and optimal control. *Procedia IUTAM*, *2*, 297–316. doi: http://dx.doi.org/10.1016/j.piutam.2011.04.027

Download on SimTK.org

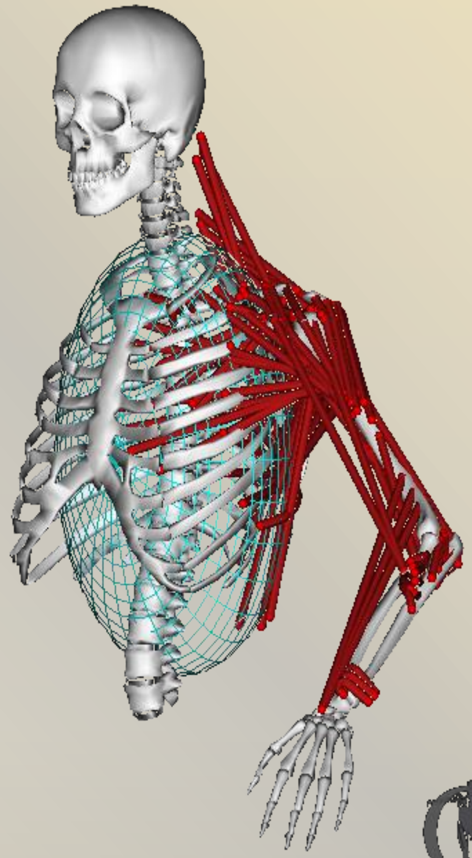
MATLAB GUI and real-time model with OpenSim viewer

(slider-based control of muscle groups)



More information and downloads

- SimTK home of the Dynamic Arm Simulator
  - https://simtk.org/home/das
- Keele Rehab group
  - http://www.keele.ac.uk/istm/rehab/
  - Google+ page
- Kirsch at the Cleveland FES Centre
- Van den Bogert at CSU



Thanks for listening!

OpenSim is supported by:

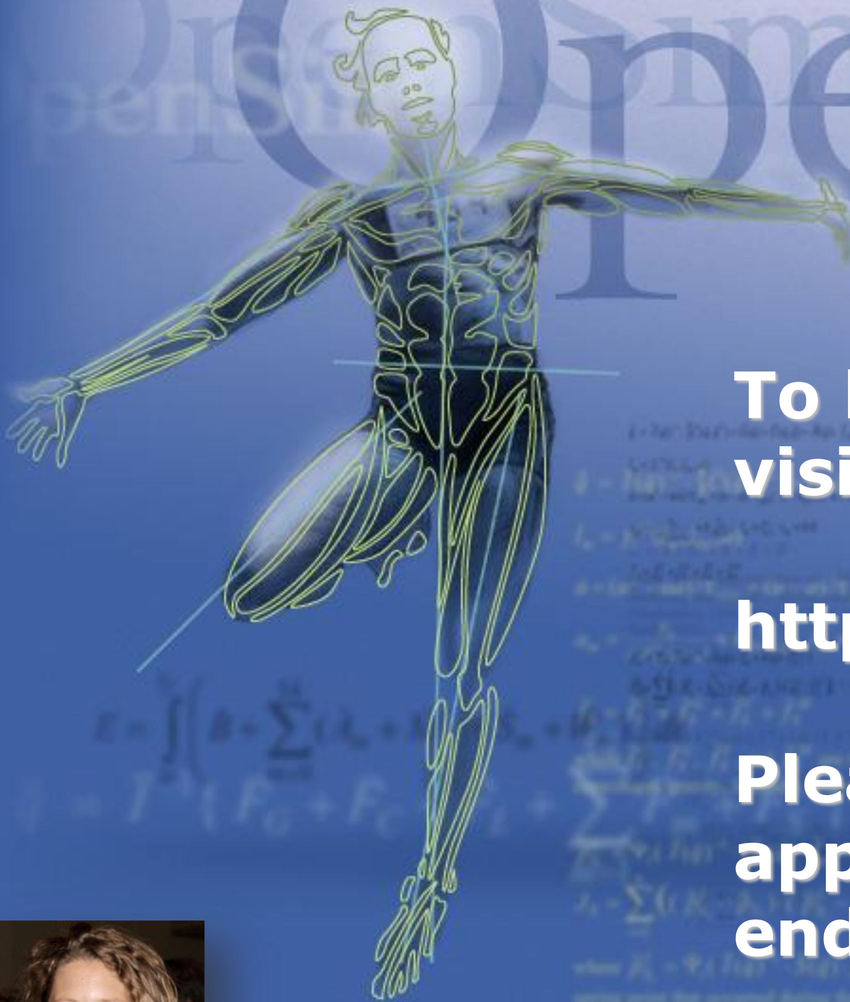
Simbios under NIH Grant U54 GM072970

The National Center for Simulation in Rehabilitation Research under NIH Grant R24 HD065690 (NICHD & NIBIB)

The NMS Physiome project funded by EU grant FP7-248189

The DARPA Warrior Web project

OpenSim



To learn more about OpenSim, visit us at:

http://opensim.stanford.edu

Please fill out the survey that appears after this webinar ends!