

SIMULATION LAB #4:

Dynamic Modeling and Simulation of Muscle-Tendon Actuators

Modeling and Simulation of Human Movement
BME 599

Laboratory Developers: Jeff Reinbolt, Hoa Hoang, B.J. Fregly, Kate Saul,
Darryl Thelen, Silvia Blemker, Clay Anderson, Scott Delp

I. Introduction

The force producing properties of muscle are complex, highly nonlinear and can have substantial effects on movement (see McMahon 1984 for review). For simplicity, lumped-parameter dimensionless muscle models, capable of representing a range of muscles with different architectures, are commonly used in the dynamic simulation of movement (Zajac, 1989). In this simulation lab, we will review the differential equations that describe muscle activation and muscle-tendon contraction dynamics when using a Hill-type muscle model. You will use OpenSim to implement muscle-tendon models and conduct simulations to investigate how various model parameters affect the dynamic response of the actuators. The lab will conclude with a *Virtual Muscle Tug-of-War* in which you will design an *optimal* muscle and compete directly against others in the class – may the best muscle win!

II. Objectives

The purpose of this lab is to learn how to use and modify OpenSim models and code to model and simulate muscle-tendon dynamics contractions.

By working through this lab, you will:

- Become familiar with the differential equations describing muscle activation and muscle-tendon contraction dynamics.
- Learn how to model and simulate dynamic musculo-tendon actions using OpenSim.
- Become comfortable with modifying existing code that models muscle activation and mechanics.
- Explore the effect of various model parameters and simulation conditions on the dynamic response of muscle.
- Design your own “optimal” muscle-tendon actuator to compete in a virtual muscle tug of war.

III. Deliverables

At the completion of this lab, you will need to turn in a written report and several OpenSim files from your modeling work.

Please turn in:

1. A written report using the report template `Simulation_Lab4_report.docx`.
2. The following files for the muscle you design (please e-mail to reinbolt@utk.edu):
 - a. YourUTNetID.osim including your muscle as muscle 1.
 - b. YourUTNetID.sto and YourUTNetID.xml with your activation history.
 - c. YourUTNetID_initialStates.sto with your initial states.

IV. Getting Started

Download the following files from the course web site:

- **Simulation_Lab4_report.docx** This file contains the template for the lab report.
- **Simulation_Lab4.osim** This file contains the template for a two-muscle tug-of-war with a block in the middle. You will use the included muscles to complete the first part of the lab. You will also use this file to help you design your own muscle for the second part of the lab and the tug-of-war competition.
- **Simulation_Lab4_Setup_Forward.xml** This file contains the parameters for running the tug-of-war simulation.
- **Simulation_Lab4_controls.sto** This storage file contains example excitations for the muscles.
- **Simulation_Lab4_states.sto** This storage file contains example initial conditions for the muscles.
- **Simulation_Lab4.mot** This movement file drives the position of the box for kinematic analysis.
- **Geometry** This folder contains the polygonal files that describe the surface of the box and floor. This folder must be in the same directory as the other files and must be named Geometry.

V. Muscle and Tendon Modeling

In this simulation lab, you will be using Hill-type musculotendon models of the form discussed in lecture. Thorough review articles have been written on the development and use of Hill-type musculotendon models in dynamic simulations of movement (Zajac 1989, Winters 1990). Following is a brief review of the Hill-type

model put forth in those papers, as it has been implemented within OpenSim. If you are interested in greater detail, you should consult the references directly.

Activation dynamics

A muscle cannot generate force or relax instantaneously. The development of force is a complex sequence of events, which begins with the firing of motor units and culminates in the formation of actin-myosin cross-bridges within the myofibrils of the muscle. When the motor units of a muscle depolarize, action potentials are elicited in the fibers of the muscle and cause calcium ions to be released from the sarcoplasmic reticulum. The increase in calcium ion concentrations then initiates the cross-bridge formation between the actin and myosin filaments. In isolated muscle twitch experiments, the delay between a motor unit action potential and the development of peak force has been observed to vary from as little as 5 milliseconds for fast ocular muscles to as much as 40 or 50 milliseconds for muscles comprised of higher percentages of slow-twitch fibers. The relaxation of muscle depends on the re-uptake of calcium ions into the sarcoplasmic reticulum. This re-uptake is a slower process than the calcium ion release, and so the time required for muscle force to fall can be considerably longer than the time for it to develop.

In the muscle simulations you will conduct in this lab, activation dynamics is modeled using a first-order differential equation with a variable time constant. This equation relates the rate of change in activation (i.e., the concentration of calcium ions within the muscle) to excitation (i.e., the firing of motor units):

$$\dot{a} = \frac{(x-a)}{\tau(a,x)} \quad (1)$$

where a is the activation level of a muscle, x is the excitation level, and τ is a variable time constant which is given by:

$$\tau(a,x) = \begin{cases} (\tau_{act} - \tau_{deact})x + \tau_{deact}, & x > a \\ \tau_{deact}, & x < a \end{cases} \quad (2)$$

In the above equation, the parameters describe the rates of rise of activation (τ_{act}) and deactivation (τ_{deact}) in response to muscle excitation. In the model, activation is allowed to vary continuously between zero (no contraction) and one (full contraction). In the body, the excitation level of a muscle is a function both of the number of motor units recruited and the firing frequency of the motor units. Some models for excitation-contraction coupling distinguish these two control mechanisms (Hatze, 1976), but it is often not computationally feasible to use such models when conducting complex dynamic simulations. In the simulation, the muscle excitation signal is assumed to represent the net effect of both motor neuron recruitment and firing frequency, and, like muscle activation, is also allowed to vary continuously between zero (no excitation) and one (full excitation). The activation and deactivation time constants can be assumed to be 15 and 50 ms, respectively (Zajac, 1989, Winters 1990).

Muscle-tendon contraction dynamics

The force producing properties of muscle are complex and nonlinear (see McMahon 1984 for review). For simplicity, a lumped-parameter dimensionless muscle model, capable of representing a range of muscles with different architectures, is commonly used in dynamic simulation of movement (Zajac, 1989). In this model, the muscle force-length and force-velocity, and tendon force-strain relationships are represented by dimensionless curves (Figure 1).

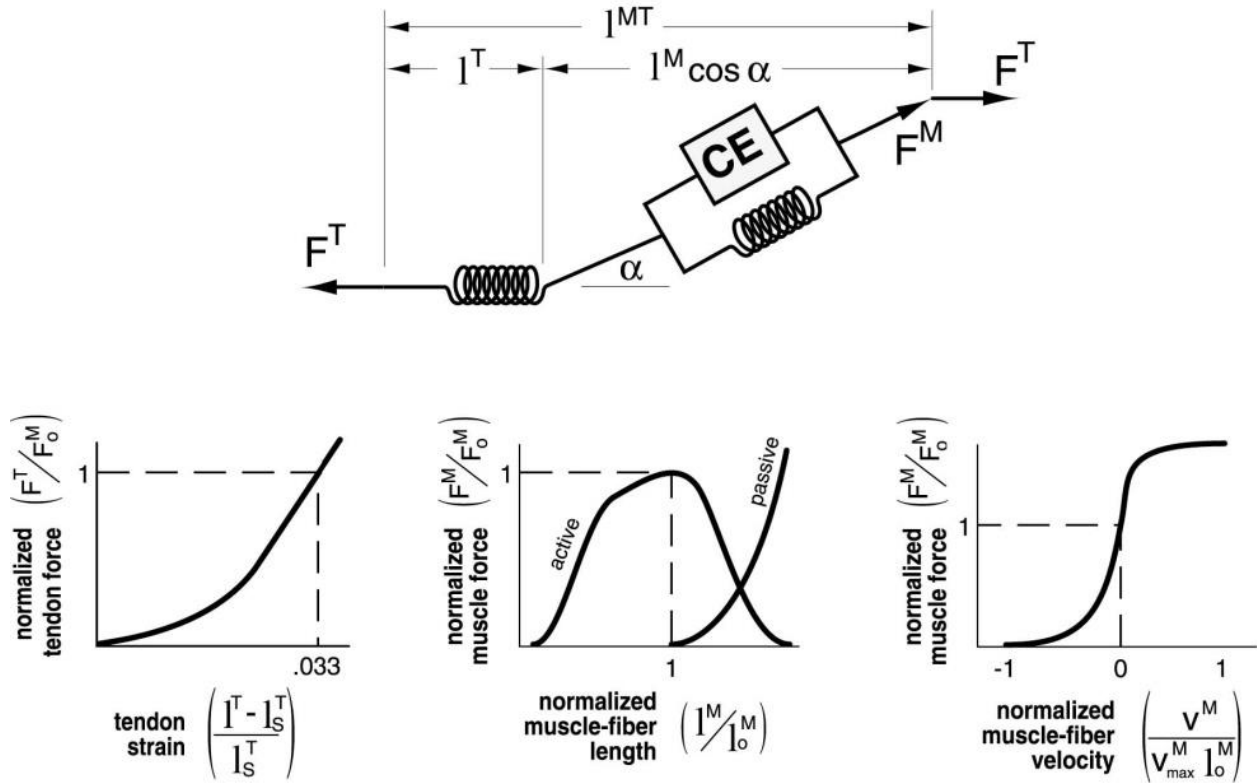


Figure 1. Dimensionless model of muscle and tendon used in our simulations. Muscle properties are represented by an active contractile element (CE) in parallel with a passive elastic element (*top*). Muscle force is dependent on muscle fiber length (*middle plot*) and velocity (*right plot*). Muscle is in series with tendon, which is represented by a nonlinear elastic element (*left plot*). Pennation angle (α) is the angle between the muscle fibers and the tendon. The forces in muscle and tendon are normalized by peak isometric muscle force (F_0^M). Muscle fiber length (l^M) and tendon length (l_s^T) are normalized by optimal fiber length (l_o^M). Tendon slack length (l_s^T) is the length at which tendons begin to transmit force when stretched. Velocities are normalized by the maximum contraction velocity of muscle (v_{max}^M). For a given muscle-tendon length (l^{MT}), velocity, and activation level, the model computes muscle force (F^M) and tendon force (F^T).

Four muscle-specific parameters are commonly used to scale the dimensionless curves for individual muscles:

- F_o^M maximum isometric muscle force
- l_o^M optimal muscle fiber length
- l_s^T tendon slack length and
- α pennation angle

There are different choices of state that can be used for the muscle-tendon model. Either muscle-tendon force (Zajac 1989, Anderson and Pandy 1999) or tendon length (Winters 1990) is commonly used. In this lab, you will be using muscle length, which has been shown to have the advantage of not requiring inversion of the tendon force-strain curve (Schutte 1992). Thus the state equation for musculotendon dynamics can be given by:

$$\dot{l}^M = v^M(\vec{q}, \dot{\vec{q}}, a, l^M) \quad (3)$$

where \vec{q} represents the generalized coordinates of the system, $\dot{\vec{q}}$ the vector of time derivatives of the generalized coordinates, a is the muscle activation and l^M is the current muscle fiber length. During a forward dynamic simulation, muscle length and activation are treated as states and solved for by numerically integrating equations (1) and (3) simultaneously with the system equations of motion. Within a simulation, the muscle-tendon force (F^{MT}) acting on the skeleton is computed from the system states: muscle length, activation, generalized coordinates and generalized speeds. The generalized coordinates and generalized speeds of the system are first used to numerically compute the muscle-tendon length (l^{MT}) and muscle-tendon velocity (\dot{l}^{MT}). For example, muscle-tendon length is computed by adding up the incremental lengths between muscle path points. Checks are included to see if wrapping surfaces are being contacted by the muscle and if so, the additional length required to wrap around a surface is included in the computation of the muscle-tendon length. Muscle-tendon velocities are computed similarly by adding up the incremental velocities between muscle path points. After computing l^{MT} and \dot{l}^{MT} , the muscle activation is used along with the dimensionless curves shown in Figure 1, to compute the muscle-tendon force.

In summary, we will be using degrees of freedom (positions and velocities), muscle length, and activation as the states of the system. Therefore, initial conditions for these quantities must be specified, and the main output of each simulation will be these states.

VI. Muscle Tug of War Model

In this lab, you will use a simple mechanical model to investigate the dynamic properties of muscle-tendon actuators. The model consists of a block translating on a frictionless surface under the action of two opposing muscle-tendon actuators, i.e., a muscle tug of war (Figure 2).

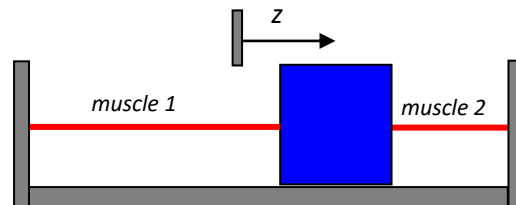


Figure 2. Model consists of a translating block on a frictionless surface being acted upon by two opposing muscles. The block is 0.1 x 0.1 m and has a mass of 20 kg. The distance between fixed supports is 0.7 meters. With the block centered, the muscle-tendon lengths are 0.3 m.

At each step, you will investigate the controls in the GUI and the structure of the files. You will be editing the files listed in the next sections.

1. Open the Simulation_Lab4.osim example.

File → Open Model... → Simulation_Lab4.osim

Use the mouse buttons to navigate the viewing window (left button rotates, middle button translates, right button zooms).

How many degrees of freedom are in this model? What are they?

What are the values of the four primary muscle parameters ($F_0^M, l_0^M, l_s^T, \alpha_0$) for each muscle?

2. Load and Run the Simulation_Lab4_Setup_Forward.xml set up file

Tools → Forward Dynamics... → Load...

Select **Simulation_Lab4_Setup_Forward.xml** and click on **Open**.

Look at the files that the setup file calls.

What do the excitations for the muscles look like? Draw them.

Press **Run** to start the simulation. You can watch the viewing window update as the simulation progresses. You can close the Forward Dynamic Settings window without affecting the simulation.

3. Plot the results

Tools → Plot...

In this section, it is important to use the files that the simulation produces for your plots and NOT the default choices within the plotter. You can use the file that appears in the list to plot state variables (block_tz, activation, and fiber length). You can load a file to investigate other output variables like muscle force, where muscle force is computed using the Actuation analysis.

Look through the files that were given to you. Where is the Actuation analysis called, and where are the results saved?

Plot box position vs. time, muscle activations vs. time, and muscle force vs. time.

You can experiment with changing muscle parameters in the GUI, saving the model, and rerunning the simulation to observe changes in the behavior of the muscles. In the next section, we'll be doing more exploration of how muscle model parameter values affect simulation results.

VII. Exploration Phase

In this section, you will gain experience modifying the files needed to run dynamic simulations. You will also begin to look at how various muscle model parameters affect the dynamic response of a muscle-tendon actuator, insight that should aid your design of your optimal muscle for the virtual tug of war.

1. For our explorations, we will use a modified tug-of-war model, with only one muscle (muscle_2). Modify the files that were given to you (save them as new files) to have a model with a single muscle at full activation starting at time 0. Make sure that file names are changed accordingly throughout to reflect your new files.

List all of the changes you had to make to change the two-muscle simulation into a one-muscle simulation, and which files were affected.

Zajac (1989) showed that the tendon-to-fiber length ratio (l_S^T/l_0^M) can have substantial effects on the mechanical response of a muscle-tendon actuator. The following sets of simulations are designed to have you vary the simulation code and parameters in order to examine the effect of tendon-to-fiber length ratio on the mechanical response of a muscle during isometric and isokinetic contractions.

2. Perform isometric (i.e., constant length) simulations with a range of tendon-to-fiber length ratios and look at how this affects the force-time response. Use the GUI or alter the .osim file to lock the block position at 0.

What is the syntax needed to lock the block in the .osim file?

Alter the tendon-to-optimal fiber length ratio for several conditions ($l_s^T/l_o^M = 0.5, 1.0, 2.0, 4.0, 8.0$). For each tendon-to-fiber length ratio, maintain the sum of the slack tendon length and optimal fiber length constant ($l_s^T + l_o^M = 0.3\text{m}$). Run the forward simulations for each case. Remember to change your initial states file accordingly.

Overlay curves of the muscle force-time histories for each tendon-to-fiber length ratio. Describe any differences that you see between the curves in terms of the rate of force development and steady state force achieved. Using what you know about muscle activation dynamics and muscle-tendon mechanics, explain why the muscle force responses differ the way they do with changes in tendon-to-fiber length ratio.

3. Perform isokinetic (i.e., constant velocity) simulations with a range of tendon-to-fiber length ratios and look at how this affects the force-position response. Load the Simulation_Lab4.mot file.

Alter the tendon-to-optimal fiber length ratio for several conditions ($l_s^T/l_o^M = 0.5, 1.0, 2.0$). For each tendon-to-fiber length ratio, maintain the sum of the slack tendon length and optimal fiber length constant ($l_s^T + l_o^M = 0.3\text{m}$). Perform a kinematic analysis by plotting tendon force vs. Simulation_Lab4 motion.

Overlay curves of the tendon force-position histories for each tendon-to-fiber length ratio. Describe any differences that you see between the shapes of the curves. Using what you know about muscle-tendon mechanics, explain why the muscle force responses differ the way they do with changes in tendon-to-fiber length ratio.

VIII. Design Phase

A muscle tug-of-war tournament will be held between members of the class. A match will consist of two muscles competing head-to-head (or muscle-to-muscle) in a one second tug of war. You will be required to specify the muscle-tendon parameters and excitation-time history subject to the constraints described below. In each tug of war, whichever muscle has moved the block onto their side at the end of one second will be declared the winner. Both muscles will start each match at rest with zero activation.

Design variables – **specify the following variables**

F_0^M	Maximum isometric muscle force
l_0^M	Optimal muscle fiber length
l_s^T	Tendon slack length
α_0	Muscle fiber pennation angle
$x(t)$	Muscle excitation time history
l^{MT}	Total muscle-tendon complex length (change ONLY z-location of muscle-P1)
τ_{act}	Activation time constant
τ_{deact}	Deactivation time constant
\bar{v}_{max}^M	Normalized maximum contraction velocity

Other model parameters you will use

A^M	Physiological cross sectional area in cm^2
V^M	Muscle volume
σ_0^M	Maximum isometric stress of muscle, assumed equal to 35 N/cm ² (Zajac 1989)

Design constraints

$F_0^M = A^M \sigma_0^M$	$F_0^M \bar{v}_{max}^M l_0^M \leq 175W$
$V^M \leq 100 \text{ cm}^3$	$\int_0^1 x(t) dt \leq 0.5$
$0.05 \text{ m} \leq l_0^M \leq 0.2 \text{ m}$	$10 \text{ ms} \leq \tau_{act} \leq 20 \text{ ms}$
$l_s^T \geq 0.1 \text{ m}$	$40 \text{ ms} \leq \tau_{deact} \leq 60 \text{ ms}$
$0.15 \text{ m} \leq l^{MT} \leq 0.4 \text{ m}$	$30 \text{ ms} \leq \tau_{deact} - \tau_{act} \leq 40 \text{ ms}$
$0^\circ \leq \alpha_0 \leq 30^\circ$	
$2 \leq \bar{v}_{max}^M \leq 10$	

You are individually responsible for devising and implementing a process for designing your muscle-tendon actuator. Document the process and results clearly in a brief report. The report, excluding figures and tables, should not be more than 2 pages. Include the following sections:

- Introduction
 - Objectives of your design.
- Methods
 - Outline of the steps used in your design process.
- Results
 - Include a combination of mathematical analysis, parameter sensitivity studies, and results of prototype muscle simulations.
 - Description of the final muscle design, including a demonstration that each criterion is met.
- Discussion
 - Justification of your final design.
 - Evaluation of the strengths and potential weaknesses of your design, e.g. under what conditions do you expect your muscle to perform well?
- References if used

IX. References

1. Anderson FC and Pandy MG (1999). A dynamic optimization solution for jumping in three dimensions. *Computer Methods in Biomechanics and Biomedical Engineering*, **2**, 201-231.
2. Hatze H (1976). The complete optimization of human motion. *Mathematical Biosciences*, **28**, 99-135.
3. McMahon TA (1984) Muscles, Reflexes, and Locomotion. Princeton University Press, Princeton, New Jersey.
4. Schutte LM (1992) Using Musculoskeletal Models to Explore Strategies for Improving Performance in Electrical Stimulation-Induced Leg Cycle Ergometry. Ph.D. Dissertation, Mechanical Engineering Department, Stanford University.
5. Winters JM (1990) "Hill-based muscle models: a systems engineering perspective," in Multiple Muscle Systems: Biomechanics and Movement Organization, edited by JM Winters and SL Woo, Springer-Verlag, New York.
6. Zajac, FE (1989) Muscle and tendon: properties, models, scaling, and application to biomechanics and motor control. *CRC Critical Reviews in Biomedical Engineering* **17**, 359-411.

Appendix A. Storage files

Create a storage file of muscle excitations (designed or from processed EMG) (all values should vary between 0 and 1). The file can be created in Excel, but must be saved as *file_name.sto*. The following headers and titles should be included:

OpenSim Header (change italicized words to match file):

```
file_name  
nRows = numberofdatarows  
nColumns = numberofdatacolumns  
endheader
```

Note: No space should be left between the header and the row of column titles

Column Titles:

```
time    nameofmuscle1.excitation    nameofmuscle2.excitation
```

Note: Muscles should be in the same order in which they appear in the force set of the .osim file.