Introduction

This page describes the Hill-type muscle models implemented in OpenSim as OpenSim::Millard2012EquilibriumMuscle by Dr. Matthew Millard and described in detail in Millard et al. (2013). Benchmarking data used for testing this model can be found at https://simtk.org/home/opensim_muscle.

Equilibrium Musculotendon Model

Musculotendon actuators consist of an active contractile element, a passive elastic element, and an elastic tendon. The maximum active force a muscle can develop varies nonlinearly with its length, represented by the active-force–length curve

\[ F \approx \frac{1}{L} \] (the tilde is used to denote forces, velocities, muscle lengths, and tendon lengths that are normalized by

\[ F_0, \dot{L}, L_m, L_t \]

, respectively). During non-isometric contractions, the force developed by muscle varies nonlinearly with its rate of lengthening, which is represented by the force–velocity curve

\[ F \approx \frac{1}{\dot{L}} \] (show error message).
Force is also developed when muscle is stretched beyond a threshold length, regardless of whether the muscle is activated, which is represented by the passive-force–length curve

Muscle force is computed using these curves as follows:

where

is the muscle activation, which ranges from to 1.

Muscle attaches to bone through tendon. Since a long tendon may stretch appreciably beyond its slack length when under tension, tendon is modeled as a nonlinear elastic element developing force according to the tendon-force–length curve.

Muscle fibers attach to tendon at a pennation angle, scaling the force they transmit to the tendon. If the tendon is assumed to be elastic and the mass of the muscle is assumed to be negligible, then the muscle–tendon force equilibrium equation (i.e.,
We have developed default force curves for the musculotendon model that have been fit to experimental data (see Millard et al., 2013 for details). These curves can be adjusted to model muscle and tendon whose characteristics deviate from these default patterns.

In a forward-dynamic simulation, the force generated by a musculotendon actuator is calculated from the length \( \ell \), velocity \( v \), and activation \( a \) of the muscle. Equation (1) alone cannot be used to solve for the muscle force because multiple combinations of muscle length and velocity will satisfy the equation. A unique solution for \( v \) can be found by solving Eq. (1) for the normalized muscle velocity \( \dot{v} \) to obtain an ordinary differential equation, which can then be integrated to simulate muscle contraction (see Zajac, 1989):

\[
\dot{v} = f(v, a) - g(v, a)
\]
is the inverse of the force–velocity curve. Although Eq. (1) is devoid of numerical singularities, Eq. (2) has four: as

\[ \text{Error rendering 'mathinline' macro: (show error message).} \]

, as

\[ \text{Error rendering 'mathinline' macro: (show error message).} \]

, as

\[ \text{Error rendering 'mathinline' macro: (show error message).} \]

, and as

\[ \text{Error rendering 'mathinline' macro: (show error message).} \]

. Since these conditions are often encountered during a simulation, the quantities causing singularities in Eq. (2) are altered so that the singularities are approached but never reached:

\[ \text{Error rendering 'mathinline' macro: (show error message).} \]

\[ \text{Error rendering 'mathinline' macro: (show error message).} \]

\[ \text{Error rendering 'mathinline' macro: (show error message).} \]

\[ \text{Error rendering 'mathinline' macro: (show error message).} \]

\[ \text{Error rendering 'mathinline' macro: (show error message).} \]

\[ \text{Error rendering 'mathinline' macro: (show error message).} \]

\[ \text{Error rendering 'mathinline' macro: (show error message).} \]

\[ \text{Error rendering 'mathinline' macro: (show error message).} \]

\[ \text{Error rendering 'mathinline' macro: (show error message).} \]

\[ \text{Error rendering 'mathinline' macro: (show error message).} \]

\[ \text{Error rendering 'mathinline' macro: (show error message).} \]

\[ \text{Error rendering 'mathinline' macro: (show error message).} \]

. Without modifying the formulation of the equilibrium model, the muscle is able to reach unrealistically short lengths (Thelen, 2003) and cannot be simulated when fully deactivated. We use a unilateral constraint on muscle length to prevent the muscle from becoming unrealistically short:
where

\[ \text{Error rendering 'mathinline' macro: (show error message).} \]

is a candidate value for

\[ \text{Error rendering 'mathinline' macro: (show error message).} \]

computed using Eq. (2).

Damped Equilibrium Musculotendon Model

The singularities in Eq. (2) arise because Eq. (1) is formulated in such a way that prevents the muscle from satisfying the equilibrium equation when it is deactivated (i.e.,

\[ \text{Error rendering 'mathinline' macro: (show error message).} \]

or when a nonzero tendon force is applied to a maximally-pennated muscle (i.e.,

\[ \text{Error rendering 'mathinline' macro: (show error message).} \]

and

\[ \text{Error rendering 'mathinline' macro: (show error message).} \]

We address these two problems by limiting the maximum pennation angle and introducing a damper (with damping coefficient

\[ \text{Error rendering 'mathinline' macro: (show error message).} \]

in parallel with the contractile element, which results in the damped equilibrium musculotendon model:

\[ \text{Error rendering 'mathblock' macro: (show error message).} \]

Since muscle length

\[ \text{Error rendering 'mathinline' macro: (show error message).} \]

is a state, Eq. (3) can be readily and uniquely solved for
using a derivative-based root-finding algorithm such as Newton's method, provided all the force development curves are $C^2$-continuous. The damped equilibrium musculotendon model of Eq. (3) should generate force profiles that are similar to those generated by the equilibrium model described by Eq. (1), but in a fraction of the simulation time because the damped equilibrium model is free of numerical singularities.

**Rigid-Tendon Musculotendon Model**

Some tendons are so stiff that they can be treated as inextensible, effectively replacing the tendon spring with an inextensible cable. The tendon inextensibility assumption is appropriate only when the tendon does not stretch sufficiently to affect the normalized length of the contractile element. This modeling simplification makes it possible to determine the muscle length ($L_m$) and velocity ($v_m$) from the musculotendon length ($L_{mt}$) and velocity ($v_{mt}$) using a kinematic model of the musculotendon actuator:

\[
\frac{dL_m}{dt} = L_{mt} - L_m
\]

Differentiating Eq. (4) with respect to time yields a relation between the muscle, tendon, and musculotendon actuator velocities:

\[
\frac{dL_m}{dt} + \frac{dv_m}{dt} = \frac{dv_{mt}}{dt}
\]

where

if the tendon is rigid. The length of the muscle (
and its orientation ( ) are coupled by a fixed-height-parallelogram pennation model:

The constant height of the parallelogram ( ) is computed using the optimal muscle length and pennation angle:

Differentiating Eq. (6) with respect to time yields an expression that can be used to calculate the pennation angular velocity:

Since the tendon length and velocity are known (i.e., ) and ( ), we use Eqs. (4) and (6) to solve for muscle length ( ) given musculotendon length ( ).
and Eqs. (5) and (7) to solve for muscle velocity (\(v_m\)) given musculotendon velocity (\(v_{mt}\)).

As with the elastic-tendon models, we use a unilateral constraint to enforce a lower bound of on muscle length. We compute the force generated by the muscle directly:

Since a muscle can generate only tensile force, we constrain this equation to remain positive:

**References**