ISB 2015

Modelling

ISB 2015-204

REAL-TIME VIRTUAL MUSCLE CONTROL FOR POWERED PROSTHESES AND EXOSKELETONS

Sandra K. Hnat* [1], Antonie van den Bogert[1]
[1]Mechanical Engineering, Cleveland State University, Cleveland, OH, United States

Preferred Presentation: Oral Presentation
If your abstract is not accepted as an oral do you wish to be considered for a poster?: Yes
Clinical Biomechanics Award: No
David Winter Young Investigator Awards: Yes
David Winter Award - presentation Preference: Oral
Emerging Scientific Award sponsored by Professor J De Luca: No
Promising Scientist Award sponsored by Motion Analysis: No

Introduction and Objectives: Powered prosthetic limbs are theoretically capable of replicating able-bodied movements. These devices are typically actuated by single-joint electric motors with time-varying linear impedance control. Though substantially better than passive prostheses, able-bodied gait is usually not achieved. We hypothesize that a more muscle-like actuation system can improve performance and simplify the feedback control. Muscles are actuators with nonlinear viscoelastic properties and internal dynamics, often crossing multiple joints. It has been shown through simulation that these properties contribute substantially to the stability of walking [1]. Here we consider the possibility of actuating a powered prosthetic system with torques generated by a real-time simulation of muscle dynamics.

Real-time simulation is challenging because the ordinary differential equations (ODE) that represent muscle dynamics are highly nonlinear and have near-singularities. Consequently, fixed step ODE solvers can become unstable and variable-step solvers will be slow due to small time steps. An alternative implicit formulation and solution method has the potential to achieve accurate results with minimal computation time [2].

The purpose of this paper is to develop and test a real-time simulation of a human muscle actuation system that generates knee and ankle torques for an above knee prosthetic device. The virtual muscle system consists of six muscles, and is driven by real-time joint angle data and muscle excitation signals.

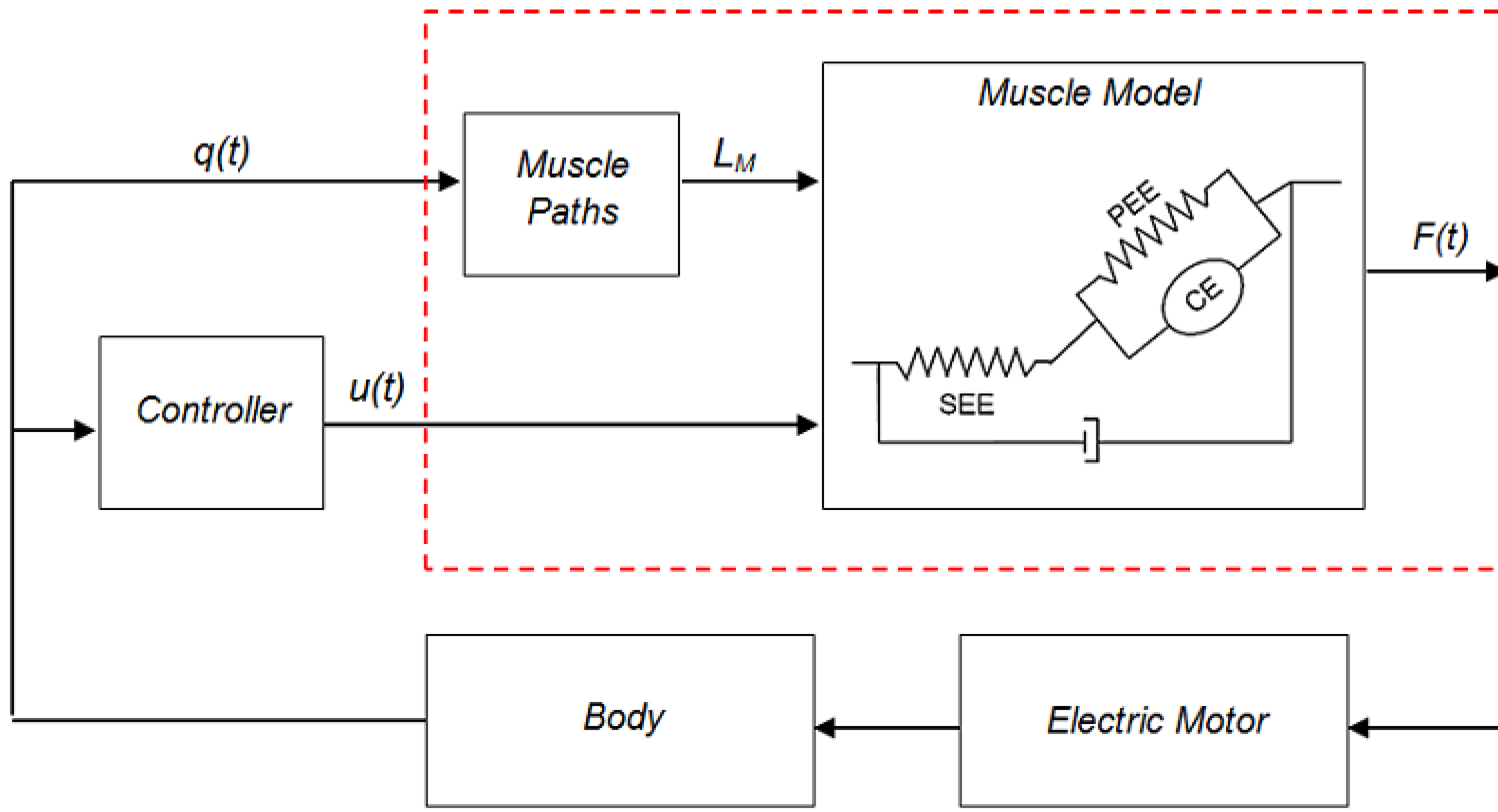
Methods: For proof of concept, we used a planar leg model with six muscles, three of which were monoarticular groups (Vasti, Soleus, Tibialis Anterior) and three were biarticular (Rectus Femoris, Hamstrings, Gastrocnemius). The inputs to the system are the joint angles (q) at the hip, knee, and ankle and the control signals (u) for the six muscles. Musculotendon length of each muscle, a required input for simulation of muscle dynamics, is computed from the three joint angles.

Muscles were represented by a Hill-type model with a contractile element (CE), series/parallel elastic elements (SEE/PEE), and a small amount of viscous damping in parallel to the contractile element. The CE has standard force-velocity and force-length properties, while PEE and SEE are modeled as nonlinear elastic. Muscle contraction dynamics was formulated as a first-order implicit differential equation (IDE). An additional first-order IDE represented the activation dynamics. The force generated by each muscle was multiplied by moment arms to obtain its contribution to the knee and ankle torques. Figure 1 is a block diagram describing the inputs and outputs of the model, in which the red box indicates the scope of this study.

Tests were performed using joint angle time histories q(t) for 30 seconds of normal walking and 1 Hz sinusoidal test signals for the muscle excitations u(t). In a prosthetic device, the former would be real-time encoder data, while the latter would be generated by a control system. Simulations were performed in Matlab with a first order Rosenbrock solver [2] using fixed time steps ranging from 0.08 to 16.0 ms. Joint torques generated by the model were compared to the result from the smallest time step in order to determine numerical error in the simulations as a percentage of the maximum moments. Computation times were measured in all tests.

Results: Muscle forces and joint torques responded appropriately to sinusoidal muscle activations. Table 1 shows how the accuracy and speed of the simulation depended on the integrator step size. RMSE was proportional to step size, due to the first order approximation used. At 0.31 ms, the simulation was just faster than real-time and the simulation errors were well below 1%. Substantially larger time steps still resulted in acceptable accuracy. For time steps larger than 10 ms, the simulation was unstable.

Figure:



Conclusion: The tests demonstrated joint torques can be generated by a real-time simulation of muscle dynamics. Computation time was well within the requirements for real-time, even in the high-level Matlab language. The muscle dynamics and differential equation solver only require simple mathematical operations and no software libraries, making them suitable for implementation on an embedded system. We conclude that this virtual muscle actuation system can be used to drive the electric motors in a prosthesis in order to better replicate the control and dynamics of able-bodied gait.

Table:

Simulation step size (ms)	RMS error in knee moment (%)	RMS error in ankle moment (%)	Solution time (s)
0.1	0.02	0.01	57.8
0.18	0.08	0.04	32.8
0.31	0.20	0.10	18.7
0.54	0.40	0.20	10.8
0.95	0.76	0.38	6.0
1.68	1.39	0.69	3.9

2.95	2.50	1.23	2.2
5.18	4.49	2.17	1.2
9.10	8.16	3.82	0.6
16.00	15.41	6.73	0.3

Caption: Integrator step size and its effect on accuracy and speed. Error was quantified as a percentage of the maximum moments in the knee and ankle.

References: [1] Gerritsen et al., Motor Control, 2: 206-220, 1998.
[2] van den Bogert et al., Procedia IUTAM, 2: 297-316, 2011.

Disclosure of Interest: None Declared