XV International Symposium on Computer Simulation in Biomechanics

July 9th – 11th 2015, Edinburgh, UK

PRIORITIZED TASK-BASED CONTROL OF MOVEMENT WITH SUPPORTING CONTACTS USING OPENSIM AND MATLAB

Misagh Mansouri[1], Vincent De Sapio[2] and Jeffrey A. Reinbolt[1]

[1] Mechanical, Aerospace, & Biomedical Engineering, University of Tennessee, Knoxville, TN, USA
[2] Information & Systems Science Lab, HRL Laboratories, LLC, Malibu, CA, USA
Email: misagh@utk.edu, Web: http://rrg.utk.edu/

INTRODUCTION

Computational synthesis of human movement is a compound task requiring accurate musculoskeletal modeling, complex dynamic simulation, and robust control system design. Moreover, this task can be better addressed by merging approaches from biomechanics and robotics. The neuro-mechanics of human movement are generally simulated by feedforward control mechanisms in the literature, where the brain controls signals to excite muscle groups that actuate joints and produce movement. However, the physiological human neuromechanical system uses feedback so that subsequent motor commands are influenced by the movement itself. Simulations accounting for such feedback control provide a robust control design for investigating human movement. In this study, we combined musculoskeletal modeling and dynamic simulation with task-based control design to synthesize human balance control without tracking of individual experimental motion analysis data. This platform has great potential for simulations predicting outcomes of surgical treatments and rehabilitations.

METHODS

We combined musculoskeletal modeling, forward dynamic simulation, and control system design to predictively synthesize human movement. This combination involved interfacing OpenSim and MATLAB software through a Simulink system function to create an environment for feedback control of musculoskeletal simulations [1]. We used OpenSim's gait model with 21 degrees-of-freedom and 92 muscle-tendon actuators in our simulations that modeled foot-ground interactions with Hunt-Crossley contact forces. We used a task-based controller in operational space [2] (Fig. 1) to provide feedback during our dynamic simulations. A task-based controller is a model-based controller relating end-effector task forces (f) to the multibody system's generalized forces (τ) through the system Jacobian (J). We mapped the joint space equations of motion into operational space using the following transformation:

$$M\ddot{q} + b + g = \tau \xrightarrow{\bar{J}^T} f = \Lambda\ddot{x} + \mu + p$$

where $\bar{J}$ is the dynamically consistent generalized inverse of the system Jacobian [3], $\Lambda(q) = (JM^{-1}J^T)^{-1}$ is the operational space mass matrix, and $\mu(q,\dot{q}) = \Lambda JM^{-1}b - \Lambda\dot{J}\dot{q}$ is the operational space centrifugal and Coriolis vector, and $p(q) = \Lambda JM^{-1}g$ is the operational space gravity vector. We used a support consistent dynamic formulation to maintain the foot-ground contact for the stance leg at all times. We defined multiple prioritized tasks to balance our musculoskeletal model on a single supporting limb in contact with the ground. The first priority task was defined as keeping the vertical projection of the whole-body center of mass (CoM) over the midpoint of the base of support (foot) using the following formulation:

$$J_{com} = \frac{1}{m_{total}} \sum_{i=1}^{n} m_i J_{com(i)},$$

where m_{total}, is the total musculoskeletal body mass, m_i is the mass of the i^{th} body, and $J_{com(i)}$ is the Jacobian of that body. The support consistent reduced Jacobian of CoM was defined as: $J^*_{com} = J_{com}\overline{SN_s}$ where, $S = [0 \quad I]$ is the actuation matrix and $N_s = I - \bar{J}_s J_s$ is the dynamically consistent null-space of the supporting contact [3]. The operational space control torque vector was then formed as:

$$\Gamma = J^*_{com}F$$

The lower priority subtasks that we defined in this study were the swing leg control task and the upper-body (torso) orientation task. Based on potential specific applications, additional tasks may be defined. For controlling these subtasks simultaneously with the main task, we controlled the null space motion and used the following compound torque formulation for task prioritization as follows:

$$\Gamma = \Gamma_{task(1)} + N^T_{task(1)}(\Gamma_{task(2)} + N^T_{task(2)}(\Gamma_{task(3)} + ...)),$$

and

$$N_{prev(k)} = I - \sum_{i=1}^{k=1} \bar{J}_{i|p(i)} J_{i|p(i)}.$$

The priority level of each task is based on where it is defined in the above equation. To solve for muscle forces, we used static optimization to minimize muscle activations squared, $min \sum a_m^2$, that satisfy the inequality condition of $0 \leq a_m \leq 1$.

To test the robustness of our controller, we introduced different movement tasks for the CoM to progress through 4 points in space while maintaining balance, not moving the swing foot, keeping the torso upright, and maintaining the foot-ground contact (Fig. 2a).

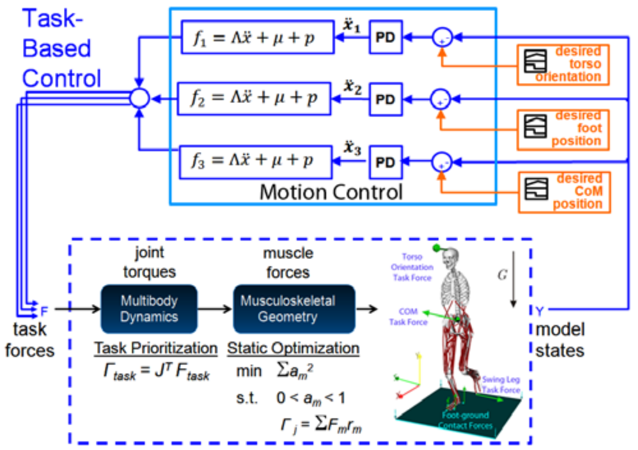

Fig 1: Schematic of the task-based neuromuscular controller implemented through the OpenSim/MATLAB interface in the Simulink environment. The interface calls the OpenSim gait model, initial controls and initial states files and performs a forward dynamics simulation with a closed-loop task-based control in operational space using the defined task forces required for performing a sequence of prioritized tasks. The task force is used to calculate the actuation force (joint torques) through the multi-body system Jacobian transpose. Muscle excitations are then calculated using static optimization.

RESULTS AND DISCUSSION

Fig. 2b shows the stance leg joint (hip, knee, ankle, and subtalar) torques (N.m) during an 8-second simulation of the assigned CoM task (progressing from point 1 through 4). The torques were then used to calculate muscle forces required for the task. During quiet standing and before the CoM task position changed (0-0.5s in Fig. 2b) the controller kept the simulated CoM within 0.1mm position error in the x, y, and z directions. This error increased to 10mm while moving CoM through its task targets.

We successfully combined a current robotics control methodology (prioritized task-based control) with state-of-the-art musculoskeletal simulations. Furthermore, this work utilized forward dynamic simulations of movement with foot-ground contact modeling that maintained the supporting contact forces in the system Jacobian calculations. In addition, the OpenSim/MATLAB interface provided the ability of interacting with musculoskeletal models along with novel control systems via feedback algorithms that may otherwise not be available to biomechanists.

Directions for future work include performing predictive simulation of human balance recovery and gait in healthy versus neurologically impaired subjects as well as validating the predictive simulation results with data collected from human subjects.

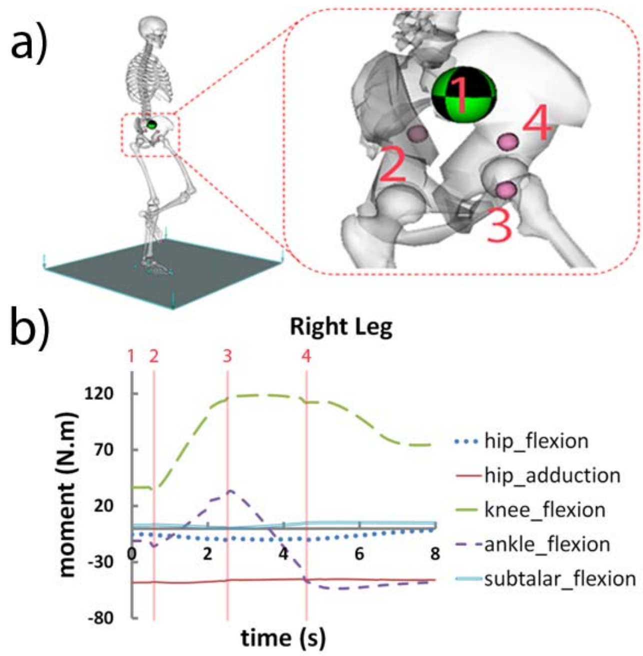

Fig 2: a) The defined task for the whole-body CoM to progress through 4 points in space while maintaining balance on one foot. b) Right leg (stance leg) joint torques required to perform the assigned task.

REFERENCES

1. Mansouri, et al., *J Biomech* **45**:1517-21, 2012.
2. De Sapio, et al., *Multibody Syst Dyn* **16**:73-102, 2006.
3. Sentis, et al., *IEEE Trans. Robotics*, **26:** 483-501, 2010.