Predictive simulation of biological movement with SCONE

Model Analysis and Modeling of Locomotion

Alice Bruel, Dimitar Stanev, Andrea Di Russo and Auke Ijspeert

{ alice.bruel – dimitar.stanev – andrea.dirusso }@epfl.ch

Start: April 23th, 2021 - Deadline: May 6th, 2021, 23:55

Introduction

The study of human locomotion has intrigued researchers to discover the principles of motor control and coordination of the limbs. Even though individual branches of neuroscience and biomechanics have produced a wealth of knowledge of the components comprising a biological motion, the fundamental question of how the neural, muscular, and skeletal systems operate together to produce efficient and purposeful movement remains largely unanswered. In this regard, the use of neuromusculoskeletal simulations can help improve our understanding.

As outlined in the previous laboratory exercise on OpenSim, inverse dynamics analysis has been used successfully to estimate quantities from motion capture recordings that are not directly observable (e.g., joint moments, muscle forces and joint loads, etc.). Even though the inverse approach has provided valuable insights into movement analysis, it relies on existing data and cannot predict new behaviors. This is where forward dynamics predictive simulations can come to the rescue. With this approach, we can simulate novel motion trajectories that perform a given task optimally, according to high-level objectives such as stability, energy efficiency, and pain avoidance without any need for experimental recordings. Predictive simulation methodologies enable researchers to pose true "what if" questions, allowing them to investigate the effects of individual models and control parameters on the simulated movement. In this exercise, we will explore SCONE, which is a free and open-source software for predictive simulation of biological motion.

Installation of SCONE

SCONE can be installed on Windows for x64 architectures[1]. Ubuntu 18.04 and Mac users can download the deb and dmg packages from the GitHub actions page[2]. The GitHub repository is configured to generate package binaries automatically using continuous integration workflows. To download the binaries, select the latest workflow run (should be green), and scroll down. We will provide the binaries so that you will use the same version of SCONE. In any case, we recommend that you make use of a Windows x64 virtual machine in case that you are not a Windows user. **When you open SCONE for the first time, navigate to *Tools -> Preferences*, then click on the *User Interface* tab. Go to the bottom and set *Perform optimization using external process* to 0.**

Deliverable

In your report, all questions should be indicated and answered as concisely as possible. All plots should be labeled with units. The submission should be named **SCONE_Name_Surname.zip** and must contain: **1) the report as a pdf file, 2) all SCONE setup files, and 3) the folder containing the optimization results**

[1] https://scone.software/

[2] https://github.com/opensim-org/SCONE/actions

removing intermediate solutions to reduce the size. Please note that we need the setup files which are copied in this folder when you run the optimization. In the GUI, you only see the .sto and .par files, but you will see other files if you navigate into the folder. By providing the folder as it is, we can quickly check your solution without evaluating the model. Regarding the optimal parameters, please include the best solution for each question so that we can reproduce your simulation.

Theoretical background

In this exercise, we will try to find the gait controller parameters (reflex-based) that produce healthy and pathological behaviors. The reflex controller defines a relationship between the muscle excitations and sensory information from proprioceptors (e.g., muscle length, velocity, and force feedback). Typically, these relationships have simple expressions such as

$$u_m(t) = c_0 + k_l(l_m(t-dt) - l_0) + k_v v_m(t-dt) + k_f f_m(t-dt)$$

where $u_m(t)$ is the excitation of muscle m at time t, $l_m(t-dt)$ the sensed muscle fiber length with dt as the sensorimotor delay, $v_m(t-dt)$ the sensed muscle fiber velocity, $f_m(t-dt)$ the sensed muscle force, and c_0, k_l, l_0, k_v, k_f are the design variables that must be determined through optimization. Each muscle reflex is active during a specific stage of the gait cycle (e.g., stance, swing). **You will not be requested to change the reflex controller in this exercise.** However, if you are interested in understanding how it works, you can find more information in Geyer and Herr 2010 work[3].

We will use a model which is a simplified version of the standard OpenSim gait model (gait2392). This model is constrained to move in the sagittal plane. It contains 9 degrees of freedom and 14 muscles. Furthermore, the model includes coordinate limiting forces at the knee to prevent joint overextension. During optimization, these forces are monitored and minimized. We include a contact model between the feet and the ground to predict the ground reaction forces that are not measured. **You can examine the provided model (Human0914.osim) in the OpenSim GUI.**

In Figure 1, we present a conceptual overview of SCONE's workflow. The controller accepts as input a set of parameters. Our goal is to find a good combination of parameters that will lead to stable and efficient locomotion. During the simulation, the controller provides a set of muscle excitations based on sensory feedback loops, and the model is numerically integrated to predict (simulate) the resulting motion (forward dynamics). Objective functions range from metabolic energy minimization to maintaining a specific speed or step length are introduced. The optimizer measures the performance of the simulation through the objective functions and adapts the design parameters. Then, a new simulation is initiated. This optimization procedure is repeated until one obtains a good solution.

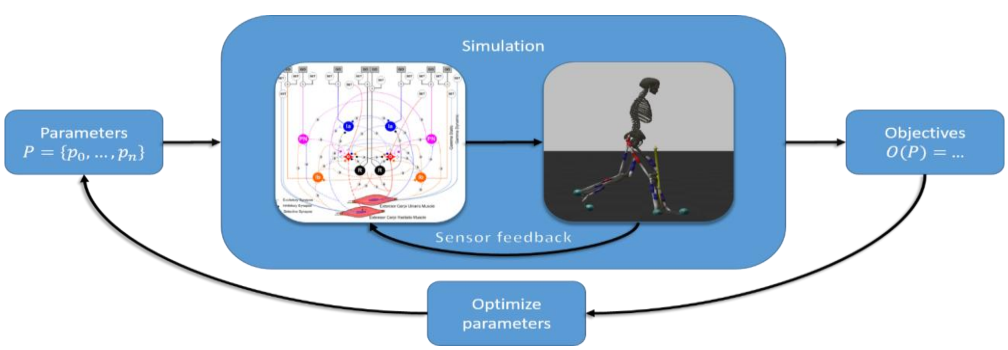

Figure 1: Conceptual overview of the single shooting method performed in SCONE.

[3]https://doi.org/10.1109/TNSRE.2010.2047592

SCONE uses the following syntax to distinguish between constant parameters and design parameters that are to be optimized:

$constant_{parameter} = value$

$design_{parameter}$ = ~$mean$ < min, max > ***or*** $mean$~std < min, max >

SCONE parses the .scone files and collects all design parameters, mean values, min and max bounds. In SCONE terminology, *Measures* monitor the performance of the simulation. Typical examples of gait measures are: cost of transport, walk with a specific gait speed, stride length or duration, muscle effort, and joint limits. You can find more information in the documentation section on the SCONE website[4].

Simulation of healthy gait

In this experiment, we will determine the reflex controller parameters that produce a healthy gait behavior for one target speed. Following the discussion mentioned above, you are provided with the following files:

1. **HealthyGait.scone**: the main script that creates the optimization and includes all files.
2. **InitialStateGait.sto**: the initial conditions for the forward dynamics numerical integration.
3. **Human0914.osim**: a planar gait model with 9 degrees of freedom and 14 muscles.
4. **ControllerComplexGH.scone:** definition of the reflex controller from Geyer and Herr with additional soleus and gastrocnemius velocity-dependent reflexes.
5. **Measure.scone**: definition of the objective functions that are to be minimized.

Deliverable 1: This task aims to perform optimization and terminate it when the gait quality is good enough. To assess the gait quality, you can do the following: 1) visual inspection of the current best solution and 2) make use of the gait analysis tool (*Windows* -> *Gait Analysis*). A good solution can be achieved for this task with less than 50 optimization generations and an objective value < 0.9. With the gait analysis tool, one can compare the joint angles and ground reaction forces against normal values.

Please comment on the results obtained from the values of the objective functions and gait analysis tool after evaluating your solution. What can be improved in terms of gait properties (please elaborate)?

Simulation of pathological gait

In these experiments, we will simulate two pathological conditions, namely heel and toe walking. Heel walking refers to a gait with exaggerated ankle dorsiflexion and can be caused by various factors resulting in weakness in the plantarflexor muscle group (e.g., gastrocnemius and soleus). This weakness can have biomechanical origins such as muscle atrophy or neural basis such as weak reflexes. On the opposite, toe walking refers to a gait with excessive ankle plantarflexion. It can also be caused by various biomechanical and neural factors, such as short or spastic plantar flexor muscle. Spasticity corresponds to an increased velocity-dependent reflex activity leading to hyperreflexia and abnormal contracture. Videos of patients walking with heel or toe conditions are provided in the material.

[4] https://scone.software/doku.php?id=doc:start

<u>Deliverable 2:</u> This task aims to model and simulate heel walking by mimicking plantarflexor muscles atrophy.

This can be achieved by reducing their maximum isometric force by including the following block of code inside the *OpenSimModel* which is located in the main .scone script file:

```
Properties {
        gastroc_l {max_isometric_force.factor = 0.5}
        soleus_l {max_isometric_force.factor = 0.5}
        ...
}
```

where the factor is multiplied by the current value (e.g., $F_o^m = factor * F_o^m$). Note that the above example modifies for the left leg only.

Please make a copy of *HealthyGait.scone* and *Measure.scone* and rename them to *Weakness.scone* and *Measure05.scone*, respectively. In *Measure05.scone*, relax the minimum target velocity to *0.5 m/s* (in pathological conditions, we expect to obtain slow-moving gaits). Do not forget to update the reference to *Measure05.scone* in *Weakness.scone* and add and complete the previous properties block to weaken the plantar flexor muscles for both feet. Try to find an appropriate value for the scaling factor to simulate the heel walking condition. More iterations (generations) might be required to obtain a satisfying result, but don't go over 200.

Please comment on the results obtained from the values of the objective functions and gait analysis tool after evaluating your solution. What are the main <u>kinematic adaptations</u> when the plantar flexors are weakened (please elaborate)?

<u>Deliverable 3:</u> The goal of this task is to model and simulate toe walking by mimicking hyperreflexia.

This can be achieved by increasing the gain of velocity-dependent reflexes of plantarflexor muscles in the controller file ControllerCompelxGH.scone:

```
# SOL and GAS V+ reflexes during stance
ConditionalController {
  states = "EarlyStance LateStance Liftoff"
  ReflexController {
    MuscleReflex {
      target = soleus
      delay = 0.020
      KV = ~1.0<0,10>  # instead of ~0.1<0,10>
    }
    MuscleReflex {
      target = gastroc
      delay = 0.020
      KV = ~1.0<0,10>  # instead of ~0.1<0,10>
    }
  }
}
```

Please make a copy of *HealthyGait.scone* and rename it to Hyperreflexia.*scone*. Again, this file should reference the *Measure05.scone* measure. Also make a copy of ControllerComplexGH.*scone* and rename it to ControllerHyperreflexia.*scone*. Make the necessary changes required to mimic hyperreflexia. Try to find an appropriate value for the reflex gain (put the same one for both muscles) to simulate the toe walking condition. Again, more iterations (generations) might be required to obtain a satisfying result but don't go over 200.

Please comment on the results obtained from the values of the objective functions and gait analysis tool after evaluating your solution. What are the main <u>kinematic adaptations</u> when hyperreflexia is introduced to the plantar flexors (please elaborate)?

<u>Deliverable 4:</u> This task aims to propose a model and simulate it to reproduce heel or toe walking.

Propose a biomechanical or neural model to reproduce heel or toe walking. You can modify another biomechanical or a neural parameter similarly to previous questions. Explain why you expect your model to result in a pathological gait.

Please make a copy of *HealthyGait.scone* and rename it to Model.*scone*. Make the necessary changes to simulate your model (if you want to modify the controller, make a copy of ControllerComplexGH.*scone* and rename it to ControllerModel.*scone).* Try to find an appropriate value for your model to simulate the pathological condition (still don't go over 200 iterations).

Please comment on the results obtained from the values of the objective functions and gait analysis tool after evaluating your solution. If your solution is not satisfying, make a hypothesis regarding eventual biomechanics or neural compensations.