# Numerical Solutions of the Equations of Motion

Andrés Cook

August 12, 2021

Given the previously mentioned difficulties with the root-finding approach to finding equilibrium configurations, my next plan was to simply solve the equations of motion directly using `ode45` until equilibrium was reached. To do this, I began with the equations of motion:

$$m\ddot{\mathbf{r}} = \mathbf{F}_{\text{net}} \tag{1}$$

$$I\ddot{\theta} = \tau_{\text{net}} \tag{2}$$

However, this leads to oscillations about equilibrium, as we are effectively applying a constant force (for small displacements, the Hill force-length function is approximately constant) to a Hookean spring. These equations will never reach a fixed value, so we obviously cannot use them to find equilibrium. However, we can introduce a damping force proportional to $\dot{\mathbf{X}}$ that will cause the oscillations to quiet down over time:

$$m\ddot{\mathbf{r}} = \mathbf{F}_{\text{net}} - \eta\dot{\mathbf{r}} \tag{3}$$

$$I\ddot{\theta} = \tau_{\text{net}} - \eta'\dot{\theta} \tag{4}$$

Note that we require a separate coefficient $\eta'$ for the rotational drag force, as $\dot{\theta}$ and $\dot{\mathbf{r}}$ have different units. This will come back to bite us later. Anyway, in the limit of very strong damping, $\eta\dot{\mathbf{r}} \gg m\ddot{\mathbf{r}}$, so the inertial term can be neglected and the drag term brought to the left-hand side. This is equivalent to the low-Reynolds' number approximation used in fluid mechanics and biophysics. Performing a similar maneuver on the rotational equation yields our final reduced-order equations of motion:

$$\eta\dot{\mathbf{r}} = \mathbf{F}_{\text{net}} \tag{5}$$

$$\eta'\dot{\theta} = \tau_{\text{net}} \tag{6}$$

As a sanity check, note that the new reduced-order equtaions have the same equilibrium points as the original equations of motion. In the code, we have set $\eta = \eta' = 1$, which is... so very wrong in so many ways. Not only are neither of those quantities dimensionless, they don't even have the same units as each other! To at least start to compensate for this, we divide the net torque by a reference distance of 1 mm, giving equations 5 and 6 the same dimensions. This is equivalent to saying that $\eta' = (1\,\text{mm})\eta$ and then setting $\eta = 1$.

Our next challenge is to determine how long to solve the equations of motion for. Fortunately, the `ode45` function allows us to specify conditions under which the algorithm will stop running, so we can simply tell it to stop when all three components of the derivative $\dot{\mathbf{X}}$ are less than some threshold. Unfortunately, sometimes the derivative just doesn't get that low. In this case, we just have to cross our fingers and hope that the time interval we solve for is long enough that the system gets close to equilibrium. To get an idea for the timescales involved in this problem, consider the case where the bar is fully horizontal and the springs are identical and fully vertical. There is then no rotation or horizontal motion and the center of mass dynamics are given by

$$\eta\dot{y} = 2ky - 2F_m \tag{7}$$

Where we double the forces to take into account the two muscles and springs, and $F_m$ is a (relatively constant) muscle force. If we take $F_m$ to be truly constant and let $y(0) = 0$), then the solution to the equation of motion is

$$y(t) = -\frac{F_m}{k}\left(1 - \exp\left(-\frac{t}{\eta/2k}\right)\right) \tag{8}$$

and the bar approaches its final position with characteristic timescale $T = \frac{\eta}{2k}$. Supposing that this timescale is within an order of magnitude for all cases, we can solve the equations of motion on the interval $t \in [0, nT]$, where n is typically of order 10-20. This method ought to give us the approximate equilibrium position, but as we will soon see, the dynamics are really much, much weirder.