Models of 1D deformable chain-like structures are important tools for computer animation [3, 6, 5]. In this assignment, we will use particle systems to model a simple chain, with an emphasis on “hard” constraints.

Consider an open chain composed of \( N+1 \) point masses, \( m = 1/(N+1)[kg] \), at positions \( x_0, x_1, \ldots, x_N \) (in meters), with adjacent masses connected by rigid (inextensible) edges of length \( h = 1/N \) [m]. We will use the Lagrangian dynamics approach of [2] (“Constrained Dynamics” chapter) to apply suitable constraints to the particle system using Lagrange multipliers (see also [1]). The “rigid edge constraint” can be modeled with the scalar constraint function,

\[
C_{RIGID}(x_i, x_{i+1}; h) = \|x_i - x_{i+1}\| - h = 0, \quad i = 0, \ldots, N.
\]

To keep the chain from falling away under gravity, use the “pin constraint”

\[
C_{PIN}(x_i; p) = \|x_i - p\| = 0
\]

to pin mass \#0 at the origin, \( p = 0 \). To make things a little more interesting, similar to [2] we’ll constrain mass \#(N+1) to lie on a ring of unit diameter using

\[
C_{RING}(x_{N+1}; p^{RING}) = \|x_{N+1} - p^{RING}\| - 0.5 = 0
\]

where \( p^{RING} = (0, -0.5, 0)^T \). Also, make the simulation run entirely in the xy plane, i.e., \( z=0 \) for all particles.

Implement a simple interactive environment to observe the dynamics of the chain, e.g., for \( N = 11 \), with the masses drawn as small spheres, and edges as cylinders. In addition to downward gravitational acceleration, \( \tilde{g} = (0, -1, 0) \) [in \( \text{m/s}^2 \)], use the cursor keys to interact with the model by applying an additional nonzero translational acceleration \( \tilde{a} \) of your choosing. Use the simple forward Euler method (or midpoint if you choose) to integrate the resulting equations of motion.

Make sure to implement Baumgarte stabilization [4] to solve the constrained equations using linear feedback control, i.e., replace each constraint equation \( C = 0 \) by

\[
0 = \dot{C} + 2b\dot{C} + b^2C.
\]

What \( b \) value works best? What happens when the damping parameter \( b \) is set too high, or too low? Can you determine \( b \) automatically? Momentarily disable \( C_{RING}^{RING} \) and drop the chain from a horizontal position: (a) compare the stabilized system \((b > 0)\) to the unstabilized one \((b = 0)\); (b) plot the error in the constraints as a function of time.

Next, try adding some simple velocity damping, \( \mathbf{f}_i = -\alpha \dot{x}_i \) to give the chain an “underwater effect.” How much damping can you add before stability becomes a problem?

In addition to submitting your code, briefly write-up your derived equations, and describe your results and answers to the previous questions. Record brief animations of your results in any convenient way.
References


