Physically-Based Simulation

Due: Feb 26
Check-In: Feb 19

Overview: Last assignment, we simulated fuzzy phenomena using dense systems of loosely interacting particles that followed some procedural set of rules for how they evolved over time. The goal of this assignment is to simulate various systems that follow well-defined physical laws. In the process, you should gain familiarity with different techniques commonly used to animate complex physical phenomena, such as differential equations, numerical integration, and spatial data-structures.

Requirements: Write (at least) two physically-based simulation demos. The first will be a simple cloth simulation. The second will either be a significantly more advanced cloth simulation, a fluid simulation, sound simulation, or deformable or rigid-body dynamics. You may work with a partner, have one partner turn in the full assignment and the other submit a note saying who they worked with.

Check-in: There is mandatory mid-way check-in. For this, you must turn in a webpage with a video of a simulated thread (or several threads) of cloth. The thread(s) must be anchored on one end and allowed to dangle free on the other end. Note, for the check-in, we only need the vertical threads – adding horizontal (cross) threads is optional, they will make the simulation less stable, so only try it second.

Strongly Suggested Features
(5) Realtime rendering (must document framerate)
(5) Video documenting your system and highlights features (e.g., working drag)
(5) 3D rendering, with user-controlled camera
(5) Real-time user interaction with system

Basic Cloth
(10) 2D Mass-spring cloth simulation
(10) 3D Mass-spring cloth simulation *[cumulative over 2D]*
(10) Drag-terms (must include demo showing effect of drag)
(10) 1-way cloth-object interaction (extra points for non-spherical objects)

Cloth Performance Benchmarks (cumulative)
(5) 15x15 Cloth at 20 FPS
(5) 20x20 Cloth at 30 FPS
(5) 30x30 Cloth with obstacle interaction at 30 FPS

General Features
Integrator
(5) Eulerian (1st order)
(5) Higher-order Explicit (e.g., Midpoint, RK4, Lax–Wendroff)
(10) Implicit Integrator
(5) Compare results of two or more integration methods

Rendering
(10) Textured simulated objects (e.g., textured cloth)

Acceleration
(5) Thread-parallel implementation (must document performance gain)
(10) SIMD / GPU implementation (must document performance gain)
(20) Spatial-data structure – must show a performance improvement

Additional Features
(10) Two-way coupling object-simulation coupling (e.g., cloth moves a ball)
(10) Integrate 2D rigid body sphere-sphere interactions for several spheres
(20) Integrate 3D rigid body sphere-sphere interactions for several spheres
(20) Combine (3D) rotational rigid body dynamics with another simulation

Advanced Cloth
(5) Wind component of drag term
(10) Tear-able cloth.
(15) Burnable cloth (should have particle effects for full credit)
(20) Self-collision in cloth
(40) Finite element Method (FEM)-based cloth simulation

Hair Simulation
(20) 2D hair simulation (can't miss collisions between strands)
(20) 3D hair simulation (can't miss collisions between strands) *[cumulative over 2D]*
(10) Angle-based dynamics (must document effect)

Deformable Objects
(20) 2D deformable objects (must show rotational effects)
(10) 3D deformable objects (must show rotational effects) *[cumulative over 2D]*

Rigid-Body Dynamics w/ Rotation
(30) Make 2D rotational rigid body dynamics simulation (10+ non-circle objects)
(60) Make 3D rotational rigid body dynamics simulation (5+ non-sphere objects)

Water Simulation
(10) 1D Water surface, with shallow water equation
(10) 2D Water surface, with shallow water equation *[cumulative over 1D]*
(30) 2D Eulerian fluid simulation (e.g. Stam GDC '03)
(50) 3D Eulerian fluid simulation

Performance Benchmarks (cumulative)
(5) 50x50 Shallow Water Sim. at 20 FPS
(5) 100x100 Shallow Water Sim. at 30 FPS
(5) 100x100 Eulerian Fluid at 20 FPS
(5) 200x200 Eulerian Fluid at 30 FPS

More things to simulate
(30) Physical simulated (polyphonic) music instrument with user interaction

Art Contest: 5 points for honorable mentions, the winner gets 10 points.

Scoring
Partial credit will be given. Scores computed as follows (points above 100 possible):
-*Undergraduate*: Grade is √(totalPoints * 100) [e.g., 100 points will be full credit]
-*Grad students*: Grade is √(totalPoints * 84) [e.g., 120 points will be full credit]

Use of other code and tools
Anything you are getting credit for must be code you wrote specifically for this course, clearly specify any external libraries you are using.

What to turn in
You must make a submission webpage with:

- Images & video of your systems
- A brief description of the features of your implementation
- Code you wrote
- List of the tools/library you used
- Brief write-up explaining difficulties you encountered
- Submission for the art contest (optional)

Remember, if you don't tell us about it, we can't give you credit for it.

Hints
-There is a lot flexibility here. If you aren't sure what to do, go for a cloth simulation that interacts with a user controlled ball. Use a spatial data-structure to accelerate cloth-ball collision detection.
-There are both easy paths and hard paths through the assignment. Don't let it become too big of a time sink, but try to have fun with it.
-Use a small timestep!
-Show off your system doing something interesting (e.g., let your cloth detach and fall onto something). It's easier to give good grades to nice simulations. =)