Report on Java Code for Cloth Simulation

Introduction

This report provides an overview and explanation of the Java code designed for simulating the behavior of a cloth-like structure in a 3D environment. The simulation involves interconnected nodes that form the cloth and their interactions with a spherical object. The code is implemented using Processing for both rendering and physics simulation.

Code Overview

The code can be divided into distinct sections, each contributing to the functionality and visual representation of the cloth simulation:

Node Class

The code starts by defining a class called `Node`. Each `Node` represents a point in the cloth grid. It has three attributes: `pos` for the 3D position, `vel` for velocity, and `last_pos` to store the previous position.

The constructor initializes a `Node` with a given position. Velocity is set to (0,0,0), and `last_pos` is initially set to the same as `pos`.

`handleSphereCollision` is a method within the `Node` class, used to detect and handle collisions between a node and a spherical object. It checks the distance between the node and the sphere and adjusts the node's position if a collision is detected.

Initialization

This part of the code encompasses the initial setup and configuration. Key elements of this section include:

Declaration and initialization of variables like `numNodes`, `Num_Rows`, `Num_Cols`, `base_pos`, as well as image loading for texturing.

Creation of two arrays: `nodes` to manage individual nodes and `nodes_cloth` to establish a 2D grid representation of the cloth.

The `nodes_cloth` grid is populated using nested loops, with each node's position calculated based on the `base_pos`, generating a grid-like structure.

Physics Parameters

The code defines critical parameters for the physics simulation:

`link_length`: Represents the length of connections between nodes.

`gravity`: Defines the force of gravity acting on the cloth.

`scene_scale`: Specifies the scaling factor for the entire scene.

`relaxation_steps` and `sub_steps` control the number of relaxation iterations and sub-steps for the simulation.

`update_physics` Function

The `update_physics` function drives the physics simulation. It involves multiple nested loops to manage node positions, velocities, and handle sphere-cloth collisions. Notably, this function is called repeatedly to update the cloth's behavior.

Input Handling

User input, such as keyboard commands, plays a role in controlling the spherical object's movements and pausing the simulation. The code includes functions for processing user keypresses and releases to enable interactive control of the simulation.

`drawSphere` Function

The `drawSphere` function is responsible for rendering the spherical object in the scene. It uses 3D graphics primitives to create and texture the sphere.

`draw` Function

The `draw` function updates the cloth's physics based on user input and manages the rendering of the spherical object and the cloth grid. The cloth grid is textured using images, and its triangular mesh is created for visualization.