CS 6323 Computer Animation and Gaming

Assignment 6 (Grade: 12 points)

Implement Inverse Kinematics

In this assignment, you are required to modify the control of the linkage from assignment 5 to implement the Inverse Kinematics (IK) using Jacobian Transpose method.

The linkage has 3 joints and 9 degree of freedoms. Each joint is associated with 3 DOF, i.e. the rotation angles along y, z, x axis, respectively.

- For any 3 DOF joint, use the rotations in the following order: y-axis, z-axis, x-axis. The initial pose vector for each bone is (0.0, 30.0, 0.0), with all numbers in degrees. The root cube object center position is (2.0, 0.5, 2.0).
- Implement the Inverse Kinematics based on Jacobian Transpose method. The end effector has 3 DOF, i.e. its position $\mathbf{e} = (e_x, e_y, e_z)$. Your program should support interactively setting the target end effector position $\mathbf{g} = (g_x, g_y, g_z)$ on GUI. The initial value of (g_x, g_y, g_z) is (3.0, 8.0. 3.0). Draw a green cube at the target position to represent it. (Grade: 1.0 points)
- The Inverse Kinematics method has the following steps:
  - While the distance between **g** and **e** is larger than a threshold (1e-6):
    - Calculate the Jacobian Matrix **J**. (Grade: 3 points)
    - Calculate the step size $\alpha = \frac{\|\mathbf{J}^{\mathrm{T}}(\mathbf{g}-\mathbf{e})\|^2}{\|\mathbf{J}\,\mathbf{J}^{\mathrm{T}}(\mathbf{g}-\mathbf{e})\|^2}$. (Grade: 1 point)
    - Update 9 DOF bone values using the transpose of **J** and step size α. (Grade: 2 points)
    - Update the end effector position **e** according to the computed 9 DOF bone values. (Grade: 2 points)

  After each iteration, please render the linkage on screen, and update the current end effector position **e** and the 9 DOF bone values on GUI. (Grade: 1 points)

- Have a start button to start the IK optimization. (Grade: 1 point)
- Have a reset button to reset the linkage to the initial state and target end effector. (Grade: 1 point)