David Schulz & Andrew Ruswick

Assignment 2

CS-4981-021

9/21/2021

**Assignment 2 – Numerical Solution**

**Forward & Inverse Kinematic Problem**

Forward and inverse kinematics involve modeling a system and finding the relationship between the actions and parameters. With forward kinematics, the position of an object is defined as a function of joint angles or speed. The inputs are the angles, and the output is the position. With inverse kinematics, you find the angles and speed to produce a given position. The input is the position, and the outputs are the angles.

**Types of Inverse Kinematics Solutions**

Analytical – Uses mathematics to find an exact answer

There are geometric and algebraic solutions

Numerical – Approximates the real solution through small update steps

For example, Euler methods for obtaining derivative

**Compare Results**

The following table shows the computed results of the analytical and numerical methods obtained from the same target positions by our code. Since our steps in the numerical method were 0.5, we could only get to the closest multiple of 0.5, but that was almost always the case. According to our forward equation, q1=194 and q2=126 are actually another correct combination of angles you can use to hit the target location of (-5,-10). If entered into the robot arm, however, this would not be possible since it is outside the hard constraints.

|  |  |  |
| --- | --- | --- |
| **Position (x,y)** | **Analytical [q1,q2]** | **Numerical [q1,q2]** |
| (0,24) | [90,0] or [90,-0] | [90,0] |
| (5,10) | [14.2,126.2] or [112.7,-126.2] | [113,-126] |
| (-5,10) | [67.3,126.2] or [165.8,-126.2] | [67,126] |
| (5,-10) | [-112.7,126.2] or [-14.2,-126.2] | [-14,-126] |
| (-5,-10) | [-165.8,126.2] or [-67.3,-126.2] | [194,126] |

**Pros & Cons**

The following are the pros and cons to each inverse kinematics method. We encountered all of them as we worked on the assignment.

Analytical

* Pros
  + Easier to code
  + Not as computationally expensive
  + Not prone to be caught by local optima
* Cons
  + Harder to understand mathematically

Numerical

* Pros
  + Easier to understand mathematically
* Cons
  + More computationally expensive
  + More complicated to code
  + Prone to be caught by local optima