Animation using Quaternions

Question

Write a function quat_slerp to perform SLERP between two quaternions and return all the intermediate quaternions.

All quaternions are represented as [Qs, Qx, Qy, Qz] in matrix form if the quaternion has scalar part Qs and vector part Qx i + Qy j + Qz k.

Input Format

- q0 is the unit quaternion representing the starting orientation, 1x4 matrix
- q1 is the unit quaternion representing the final orientation, 1x4 matrix
- steps is the number of quaternions required to be returned, integer value

Output Format

- The first step is q0, and the last step is q1
- q_int contains q0, steps-2 intermediate quaternions, q1 - in the order of ascending order of timesteps
- q_int is a (steps x 4) matrix

Code

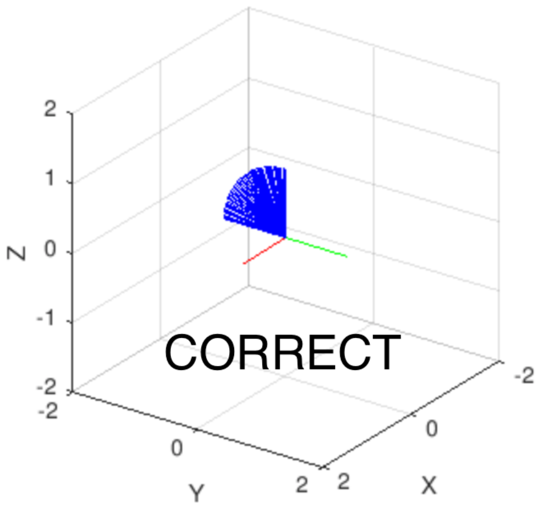
- Write your code in the space provided within the script, as per the script comments.

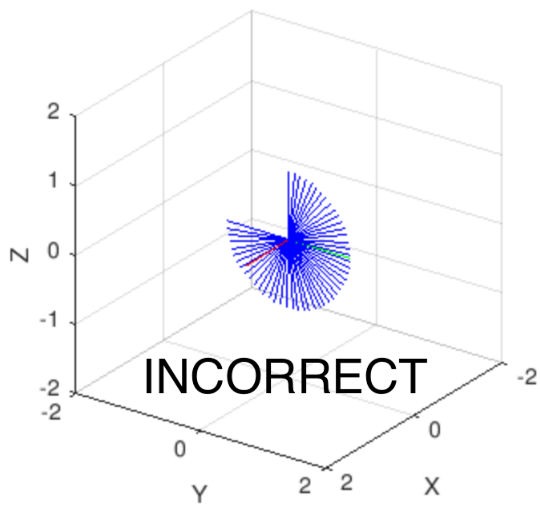
Helpful Notes

- Since the quaternions are unit quaternions, we can find the angle between them using dot product.
- $cos(\Omega) = q_o.q_1$
- At time-step t, the intermediate quaternion is given as:

$$\frac{sin[(1-t)\Omega]}{sin(\Omega)} q_0 + \frac{sin[t\Omega]}{sin(\Omega)} q_1$$

- **IMPORTANT:** We are looking for the shortest path between two rotations, as depicted in the figures below.





- Make use of visualization scripts available here - Visualise SLERP - to check if your solution is coming along correctly. The plots in the images above were generated using such scripts.

Additional Notes

Consider the case where we are animating a rotation from q_0 to q_1. The 'Visualize SLERP' code provided to you shows you the following:

The world coordinate frame is the X-Y-Z frame as per Matlab's plot. The solid X-Y-Z lines is the coordinate frame obtained by rotating the world axis by q_0 - This is the starting orientation for the animation. The dashed X-Y-Z lines represent the coordinate frame obtained by rotating the world axis by q_1 - This is the final orientation for the animation. The code then animates another set of dashed X-Y-Z lines from the starting coordinate frame to the final coordinate frame.

Now coming to your question on the angle obtained from the 'dot' product. Consider the equation $q_1 = q_0 * q$. Here q is the quaternion that which rotates from q_0 to q_1. Then pre-multiplying both sides by q_0^{-1} we get

$$q_0^{-1} * q_1 = q$$

If $q = [\cos(\Omega), \vec{r} \sin(\Omega)]$, by equating the scalar parts of the above equations, you will see that $\cos(\Omega) = q_0.q_1$, the 'dot' product of the quaternions.

So Ω is half the angle of rotation between q_0 and q_1, about the axis $\vec{r}$.

Hope that makes things clear!