Homework 2

EECS/BioE/MechE C106A/206A
Introduction to Robotics

Due: September 13, 2022

Note: Problems marked [bonus] will be eligible for a (very) small amount of extra credit, though you cannot receieve more than a full score on the homework as a whole. We encourage you not to spend exorbitant amounts of time on these questions, and as such, you may receieve partial credit for attempting them.

Note 2: This problem set includes two programming components. Your deliverables for this assignment are:

1. A PDF file submitted to the `HW2 (pdf)` Gradescope assignment with all your work and solutions to the written problems.
2. The provided `kin_func_skeleton.py` and `hw2.py` file submitted to the `HW2 (code)` Gradescope assignment with your implementation to the programming components. Make sure to select both files when submitting your assignment.

Problem 1. Exponential Coordinates for Rotations

Recall that for any rotation matrix $R \in SO(3)$, there exists a unit axis vector $\omega \in \mathbb{R}^3$, a corresponding skew symmetric matrix $\hat{\omega} \in \mathfrak{so}(3)$, and a scalar θ such that $R = e^{\hat{\omega}\theta}$. Further recall the geometric interpretation of exponential coordinates; to write $R = e^{\hat{\omega}\theta}$ is to state that R implements a rotation about the unit axis ω by θ radians in the positive direction (according to the right hand rule). (Also, while it is not necessary for this problem, recall that the exponential is derived from solving a differential equation relating the angular velocity of a point and the axis: $v = \dot{q} = \omega \times q(t)$, $q(t) = R = e^{\hat{\omega}t}q_0$)

(a) Let $\omega = (\omega_1, \omega_2, \omega_3)^T \in \mathbb{R}^3$ be a unit vector and recall that we define the hat operator as

$$\hat{\omega} = \begin{bmatrix} 0 & -\omega_3 & \omega_2 \\ \omega_3 & 0 & -\omega_1 \\ -\omega_2 & \omega_1 & 0 \end{bmatrix} \tag{1}$$

Note that we denote this operator as either $\hat{\omega}$ or $\omega^\wedge$ interchangeably. Further, we define the "vee" operator $^\vee$ as the inverse of hat, so that $\hat{\omega}^\vee = \omega$. "vee" is defined on $\mathfrak{so}(3)$ and returns a 3-vector.

Let $\theta \in [0, \pi]$ be a scalar. Show that the matrix $\hat{\omega}\theta$ has eigenvalues $\{0, i\theta, -i\theta\}$.

(b) Let R be the rotation matrix for which (ω, θ) is a set of exponential coordinates. i.e. $R = e^{\hat{\omega}\theta}$. Find the eigenvalues of R.

Hint: Recall the properties of the matrix exponential we introduced in Homework 0.

(c) For what values of the rotation angle θ does R have 1 or 2 distinct real eigenvalues? Can it ever have 3 distinct real eigenvalues?

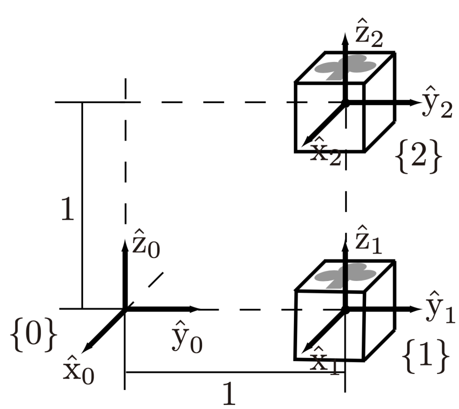
Hint: Recall Euler's formula.

(d) Interpret your answer to part (c) geometrically. When R has exactly 1 real eigenvalue, what is it and what is the corresponding eigenvector? Why does this make sense geometrically given that R is a rotation matrix? What about when R has two distinct real eigenvalues? You should answer this question without ever carrying out a direct eigenvector computation.

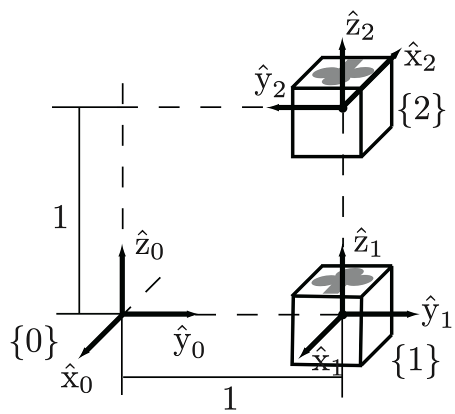
Problem 2. Finding Exponential Coordinates

In each of the following subparts, find the exponential coordinates of the rigid body transform requested.

(a) Figure (1) shows a cube undergoing two different rigid body transformations from frame $\{1\}$ to frame $\{2\}$. In both cases, find a set of exponential coordinates for the rigid body transform that maps the cube from its initial to its final configuration, as viewed from frame $\{0\}$. Do this by first finding the equivalent screw motion.



(a) A first screw motion.



(b) A second screw motion.

Figure 1: A cube undergoing two screw motions.

(b) For a point $p_0 \in \mathbb{R}^3$, consider a following rigid body motion in which the velocity of the point is

$$\dot{p}(t) = \omega \times (p(t) - q), \quad p(0) = p_0, \tag{2}$$

where $\omega = [0,0,1]^T \in \mathbb{R}^3$ is the axis of rotation and $q = [1,1,1]^T \in \mathbb{R}^3$ is the center of the rotation. $p(t)$ is a coordinate of the point at time t with respect to the frame 0. This is depicted in Figure (2). If $g(t) \in SE(3)$ is a 4×4 matrix such that

$$\begin{bmatrix} p(t) \\ 1 \end{bmatrix} = g \begin{bmatrix} p_0 \\ 1 \end{bmatrix}. \tag{3}$$

Find a $\xi = (v, \omega) \in \mathbb{R}^3$ such that $e^{\hat{\xi}t} = g(t)$.

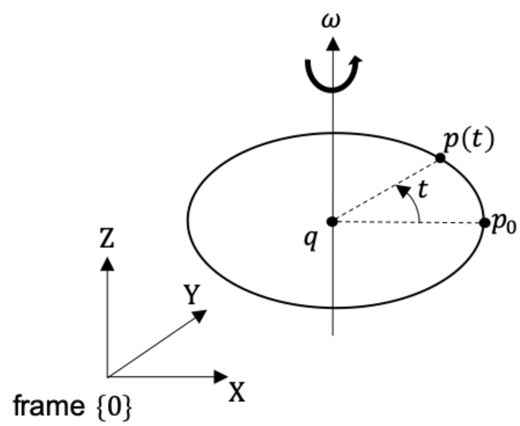


Figure 2: The coordinate of the moving point is $p(t)$. q is $[1,1,1]^T$ and ω is $[0,0,1]^T$.

(c) Let $g = (R, p) \in SE(3)$, with $R = R_x(\pi/2)R_z(\pi)$, and $p = (0, -1/\sqrt{2}, 1/\sqrt{2})$. Find the exponential coordinates of g.

Hint: Try drawing out the transformation between the initial and final frames after applying g, and attempt to find an equivalent screw motion. You shouldn't need to solve this algebraically.

Problem 3. Implementing Exponential Coordinates

What good is all this theory if we can't use it for something? In order to see the applications of the exponential map, we'll first need to implement a few fundamental equations in code. Fill in the provided `kin_func_skeleton.py` file to implement the following formulas using numpy. Test your implementation with the provided test cases by simply running `python kin_func_skeleton.py` in the command line. You will need this code to start Lab 3.

(a) The "hat" $(\cdot)^\wedge$ operator for rotation axes in 3D.

– Input: 3×1 vector, $\omega = [\omega_x, \omega_y, \omega_z]^T$

– Output: 3×3 matrix,

$$\hat{\omega} = \begin{bmatrix} 0 & -\omega_z & \omega_y \\ \omega_z & 0 & -\omega_x \\ -\omega_y & \omega_x & 0 \end{bmatrix} \tag{4}$$

(b) Rotation matrix in 3D as a function of ω and θ

– Input: 3×1 vector, $\omega = [\omega_x, \omega_y, \omega_z]^T$ and scalar, θ

– Output: 3×3 matrix,

$$R(\omega, \theta) = e^{\hat{\omega}\theta} = I + \frac{\hat{\omega}}{\|\omega\|} \sin(\|\omega\|\theta) + \frac{\hat{\omega}^2}{\|\omega\|^2}(1 - \cos(\|\omega\|\theta)) \tag{5}$$

(c) The "hat" $(\cdot)^\wedge$ operator for Twists in 3D.

– Input: 6×1 vector, $\xi = \left[v^T, w^T\right]^T = [v_x, v_y, v_z, \omega_x, \omega_y, \omega_z]^T$

– Output: 4×4 matrix,

$$\hat{\xi} = \begin{bmatrix} \hat{\omega} & v \\ 0 & 0 \end{bmatrix} = \begin{bmatrix} 0 & -\omega_z & \omega_y & v_x \\ \omega_z & 0 & -\omega_x & v_y \\ -\omega_y & \omega_x & 0 & v_z \\ 0 & 0 & 0 & 0 \end{bmatrix} \tag{6}$$

(d) Homogeneous transformation in 3D as a function of twist ξ and joint angle θ.

– Input: 6×1 vector, $\xi = \left[v^T, w^T\right]^T = [v_x, v_y, v_z, \omega_x, \omega_y, \omega_z]^T$ and scalar θ

– Output: 4×4 matrix,

$$g(\xi, \theta) = e^{\hat{\xi}\theta} = \begin{cases} \begin{bmatrix} I & v\theta \\ 0 & 1 \end{bmatrix} & w = 0 \\ \\ \begin{bmatrix} e^{\hat{\omega}\theta} & \frac{1}{\|w\|^2}\left(\left(I - e^{\hat{\omega}\theta}\right)(\hat{\omega}v) + \omega\omega^T v\theta\right) \\ 0 & 1 \end{bmatrix} & \omega \neq 0 \end{cases} \tag{7}$$

(e) Product of exponentials in 3D.

– Input: n 6D vectors, $\xi_1, \xi_2, \ldots, \xi_n$ and scalars, $\theta_1, \theta_2, \ldots, \theta_n$

– Output:

$$g\left(\xi_1, \theta_1, \xi_2, \theta_2, \ldots, \xi_n, \theta_n\right) = e^{\hat{\xi}_1\theta_1} e^{\hat{\xi}_2\theta_2} \ldots e^{\hat{\xi}_n\theta_n} \tag{8}$$

Problem 4. Satellite System

Two satellites are circling the Earth as shown in Figure 3. Frames $\{1\}$ and $\{2\}$ are rigidly attached to the satellites in such a way that their $\hat{x}$-axes always point toward the Earth. Satellite 1 moves at a constant speed v_1, while satellite 2 moves at a constant speed v_2. To simplify matters, ignore the rotation of the Earth about its own axis. The fixed frame $\{0\}$ is located at the center of the Earth. Figure 3 shows the position of the two satellites at $t = 0$. For the following questions, you may leave your answers in terms of the products of *known* matrices.

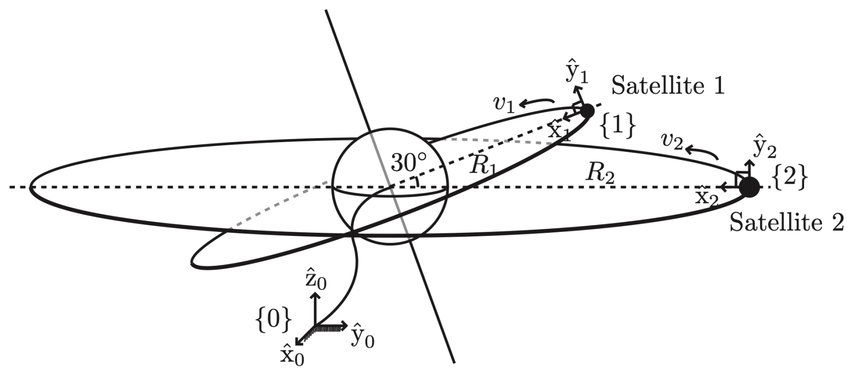


Figure 3: Two satellites circling the Earth. In both cases, the satellite's z-axis points directly into the page (tangent to the orbit).

(a) Derive the frame g_{02} at time $t = 0$ as a 4×4 homogeneous transform matrix.

(b) Derive the frame g_{02} as a function of t as a 4×4 homogeneous transform matrix. *Hint: See if you can determine the time-dependent transform from frame 2's configuration at time t to its initial configuration, and then apply part (a)*

(c) Derive the frame g_{01} as a function of t as a 4×4 homogeneous transform matrix. *Hint: Does the motion of satellite 1 looks similar to that of satellite 2? How are they different?*

(d) Using your results from part (b) and (c), find g_{21} as a function of t.

(e) At this point, you may be frustrated with the amount of work it took to explicitly define the movement of an object undergoing a simple circular motion in 3d space. Instead of explicitly writing out the 4×4 homogenous transform as a function of time, we can instead note the *twist* associated with the rigid body motion, a 6×1 vector which can be "hatted" to create exponential coordinates. For this problem, find the twist ξ_2 such that the twist takes on the exponential coordinates for motion of satellite 2, or in other words, ξ_2 satisfies

$$g_{02}(t) = e^{\hat{\xi}_2 t} g_{02}(0)$$

Hint: You should not have to take any matrix logarithms here. Think about what each element of ξ represents.

(f) Similar to the previous part, find the twist ξ_1 such that ξ_1 satisfies

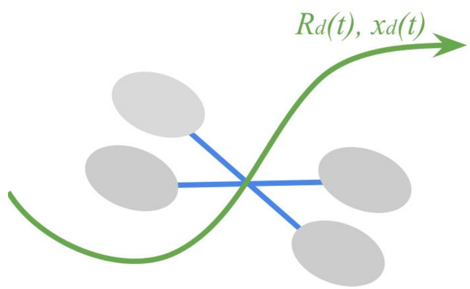
$$g_{01}(t) = e^{\hat{\xi}_1 t} g_{01}(0)$$

(g) Fill in the corresponding parts of `hw2.py` to implement your answers to parts (b)-(e) above. Note that your credit for this problem will be awarded by the autograder configured to the `HW2 (code)` assignment on Gradescope.

You can visualize the motion of these frames by running the `g_t_vis.py` and `xi_vis.py` after completing the relevant sections of `hw2.py`. Note that both `g_t_vis.py` and `xi_vis.py` will only work after filling out `kin_func_skeleton.py`, and `xi_vis.py` needs all parts of `hw2.py` completed. Use your scroll wheel to zoom camera, ctrl+drag to rotate camera, and shift+drag to pan camera. This may be useful for verifying your computations before submitting to Gradescope, and fun to play with as well. What cool rigid body motions can you come up with?

Problem 5. Bonus: Close Encounters of the SO(Third) Kind

Quadcopter UAVs are one of the most exciting and quickly developing fields in robotics. In this problem, we'll explore how rotation matrices are used in their control.



When controlling a quadcopter, we wish to be able to control both the position ($x \in \mathbb{R}^3$) and the orientation ($R \in SO(3)$) of the quadcopter. We may acheive this control by using a *feedback control system*, something we'll discuss later in the course!
To design a feedback controller, we must be able to measure the distance between the current state of our system and the state we'd like it to be at, known as the *desired state*.
For position, this is simple to define. If we'd like our quadcopter to be at a position $x_d \in \mathbb{R}^3$, but it's currently at a position $x \in \mathbb{R}^3$, we define the *position error* as follows:

$$e_x = x_d - x \tag{9}$$

How might we find the distance between two orientations in space? To do this, we must define a function that enables us to find the difference between two rotation matrices: the desired rotation matrix, R_d, and the current rotation matrix of the quadcopter, R. Because these are matrices, not vectors, subtracting the two won't tell us the distance between them! In the paper *Geometric Tracking Control of a Quadrotor UAV on* $SE(3)$, (a must-read for anyone enthusiastic about quadrotor control!) Taeyoung Lee et al. used the following function as a measure of distance between rotation matrices:

$$\Psi(R, R_d) = \frac{1}{2}\mathrm{tr}[I - R_d^T R] \tag{10}$$

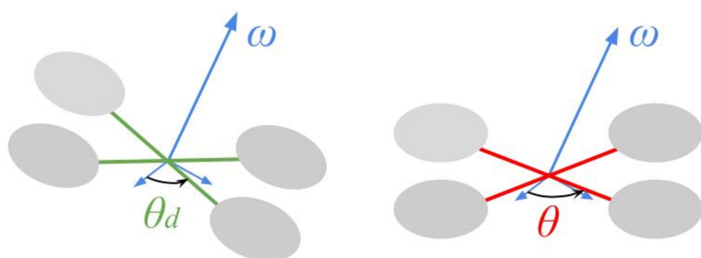
This is known as an *error function on* $SO(3)$. Let's explore some of its properties, and see why it's a good choice for quadrotors.

(a) First, we want to make sure that when we're at the correct orientation, there will be zero error, and that the only time error is zero is when we're at that orientation.

Prove that the error function $\Psi(R, R_d) = 0$ if and only if $R = R_d$. *Hint: Recall that to prove an "if and only if" statement, we must show it is true in both directions of implication.*

(b) Before we tackle some more challenging properties of the configuration error function, we'll need a few more intermediate results.
Prove that for nonzero $\omega \in \mathbb{R}^3$, $\hat{\omega}^2 = \hat{\omega} \cdot \hat{\omega}$ is negative semidefinite, where $\wedge : \mathbb{R}^3 \to so(3)$ is the hat map. *Hint: A matrix $M \in \mathbb{R}^{n \times n}$ is negative semidefinite if it is symmetric and $x^T M x \leq 0$ for all nonzero $x \in \mathbb{R}^n$.*
Remark: If a matrix is negative semidefinite, all of its eigenvalues are less than or equal to 0. Keep thinking about its eigenvalues for the rest of the problem!

(c) Let $R_d = e^{\hat{\omega}\theta_d}$ and $R = e^{\hat{\omega}\theta}$, where $\omega \in \mathbb{R}^3$, $||\omega|| = 1$, and $\theta,\ \theta_d \in \mathbb{R}$. Assume ω is fixed and does not change with time.
Prove that for these rotation matrices, the local extrema of Ψ occur at $\theta - \theta_d = n\pi$, where n is an arbitrary integer. Explain why it intuitively makes sense that the extrema of the function occur at these angles. *Hints: use Rodrigues' Formula, recall that* $\mathrm{tr}(A + B) = \mathrm{tr}(A) + \mathrm{tr}(B)$, *and think about applying the remark from part (b)!*



Above: The green quadrotor is at the desired angular configuration, while the red quadrotor is at an angle θ with respect to the axis of rotation.

(d) Let $R_d = e^{\hat{\omega}\theta_d}$, $R = e^{\hat{\omega}\theta}$, and $\omega = [0, 0, 1]$. Sketch the error function $\Psi(R, R_d)$ as a function of $\theta - \theta_d$ on the domain $[0, 2\pi]$ for these two rotation matrices. Is this function continuous on this domain? Is it differentiable with respect to $\theta - \theta_d$ on this domain?