1. Orthogonal transformations

Recall that an $n \times n$ matrix $A$ is an *orthogonal transformation* if any of these four equivalent criteria are met:

- $A$ has $n$ mutually perpendicular, unit length rows or columns
- the rows or columns of $A$ form an orthonormal basis for $\mathbb{R}^n$
- $AA^T = A^T A = I$
- $A^{-1} = A^T$

Also recall that orthogonal transformations preserve inner products. For any orthogonal transformation $A$ and any two vectors $x$ and $y$,

$$(Ax) \cdot (Ay) = x \cdot y$$

Using the definition and property above, prove the following:

a. The product of any two orthogonal transformations $A$ and $B$ is itself an orthogonal transformation.

b. For any orthogonal transformation $A$ and any vector $x$,

$$\|Ax\| = \|x\|$$
2. Rigid transformations

a. As we said in class, a rigid transformation in 2D parameterized by a rotation matrix $R$ and a translation vector $t$ can be represented by the 3x3 matrix

\[ M = \begin{bmatrix} R & t \\ 0 & 1 \end{bmatrix} \]

Show that the product of two such matrices $M_2$ and $M_1$ parameterized by $R_2$, $t_2$, $R_1$, and $t_1$, results in a matrix which is equivalent to the composition of the rigid transformations that we discussed in class.

b. A pair of transformations $T_1$ and $T_2$ is said to obey the *commutative property* if

\[ T_2(T_1(p)) = T_1(T_2(p)) \]

For each of the following, show that the pair of transformations commutes, or provide a counterexample indicating that they do not.

- any two rotations in $\mathbb{R}^2$
- any two translations in $\mathbb{R}^n$
- any two rigid transformations in $\mathbb{R}^2$
- any two rotations in $\mathbb{R}^3$
3. Simulating robot motion

A differential drive robot has a wheel radius of 0.05 m, and the wheel centers are each a distance of 0.05 m from the centerline (so the wheel centers are 0.1 m apart). Starting out at \((x_w = 0, y_w = 0, \theta = 0)\), the robot’s wheels undergo the following velocities:

- \(v_L = 1.5, v_R = 2.0\) for 3 seconds
- \(v_L = 1.0, v_R = -1.0\) for 2 seconds
- \(v_L = 0.5, v_R = 2.5\) for 2 seconds
- \(v_L = -2.0, v_R = 1.0\) for 3 seconds

Write a Python program to simulate the motion of the robot using the equations from class. You should assume that the robot changes its wheel velocities instantaneously.

a. Run your program three times, with step sizes of \(\Delta t = 0.5\) s, 0.25 s, and 0.125 s.

b. Use \texttt{matplotlib} to create an \(x/y\) plot of all three robot paths on a single graph. Use different point/line styles to distinguish the paths, and make sure to use an equal aspect ratio when plotting (that’s \texttt{axis('equal')}, in \texttt{matplotlib}), otherwise your circles will look like ovals.

c. Note what happens as the timestamp gets smaller. Are there systematic errors caused by large step sizes, or is the error more or less random? Write your answers on the same page as your plot printout.