Homework 1

Yongxin Wang
ywang51@mason.gmu.edu

February 1, 2017

1. Mason Web page: **https://mason.gmu.edu/~ywang51/CS685**

2. Solution

   a)

$$\det(T_1) = \begin{vmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{vmatrix} = \cos^2\theta - (-\sin^2\theta) = 1 \tag{1}$$

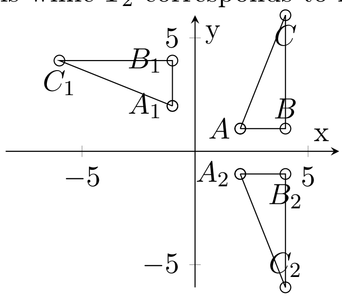
   b) Yes. It is orthonormal because $\boldsymbol{T}_1^\top = \boldsymbol{T}_1^{-1}$.

$$\boldsymbol{T}_1\boldsymbol{T}_1^\top = \begin{bmatrix} \sin\theta & \cos\theta \\ \cos\theta & -\sin\theta \end{bmatrix}\begin{bmatrix} \sin\theta & \cos\theta \\ \cos\theta & -\sin\theta \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} = \boldsymbol{I} \tag{2}$$

$$\det(\boldsymbol{T}_2) = \begin{vmatrix} \sin\theta & \cos\theta \\ \cos\theta & -\sin\theta \end{vmatrix} = -\sin^2\theta - \cos^2\theta = -1 \tag{3}$$

   c)
   - Yes. T_2 is rigid body transformation because it maintains the relative position relationship between every two points in the original object.
   - T_1 corresponds to rotations while T_2 corresponds to reflections.

   Take $\theta = 90°$ as example,



3. Solution

It's fixed angles rotation. So left matrix multiplication operation applies in this circumstance. Here, c stands for cos; s stands for sin.

$$\boldsymbol{R} = \boldsymbol{R}_{X_A}(\phi)\boldsymbol{R}_{Z_A}(\theta) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & c\phi & -s\phi \\ 0 & s\phi & c\phi \end{bmatrix} \begin{bmatrix} c\theta & -s\theta & 0 \\ s\theta & c\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} c\theta & -s\theta & 0 \\ c\phi s\theta & c\phi c\theta & 0 \\ s\phi s\theta & s\phi c\theta & 0 \end{bmatrix} \tag{4}$$

4. Solution

a) Given that the value of θ is always between 0 and 180 degree, we compute the following values.

$$\theta = \arccos\left(\frac{r_{11} + r_{22} + r_{33} - 1}{2}\right) = \arccos(-0.24065) = 103.9^\circ \tag{5}$$

$$\hat{\boldsymbol{\omega}} = \frac{1}{2\sin\theta}\begin{bmatrix} r_{32} - r_{23} \\ r_{13} - r_{31} \\ r_{21} - r_{12} \end{bmatrix} = \frac{1}{2\sqrt{1-\cos^2\theta}}\begin{bmatrix} 0.9739 - (-0.1468) \\ 0.9739 - (-0.1468) \\ 0.9739 - (-0.1468) \end{bmatrix} = \begin{bmatrix} 0.5773 \\ 0.5773 \\ 0.5773 \end{bmatrix} \tag{6}$$

b) Result from matlab function **eig**,

$$\boldsymbol{V} = \begin{bmatrix} -0.2887 + 0.5\boldsymbol{i} & -0.2887 - 0.5\boldsymbol{i} & 0.5774 \\ 0.5774 & 0.5774 & 0.5774 \\ -0.2887 - 0.5\boldsymbol{i} & -0.2887 + 0.5\boldsymbol{i} & 0.5774 \end{bmatrix} \tag{7}$$

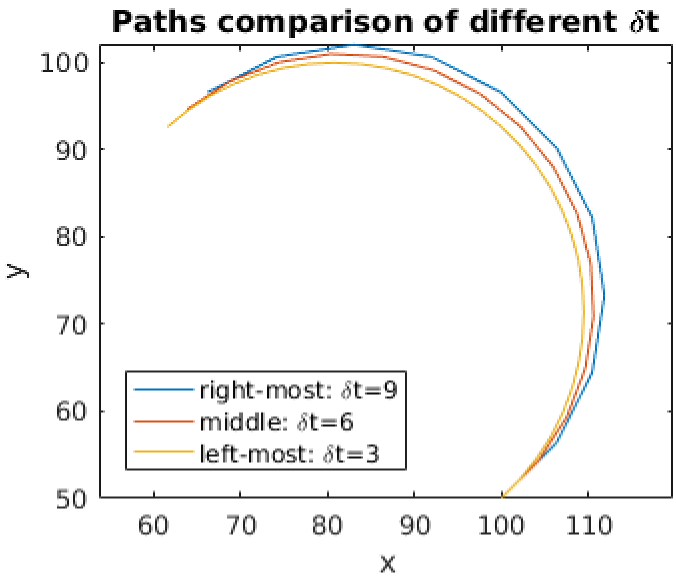
$$\boldsymbol{D} = \begin{bmatrix} -0.2406 + 0.9706\boldsymbol{i} & 0 & 0 \\ 0 & -0.2406 + 0.9706\boldsymbol{i} & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{8}$$

We noticed here that the eigenvector associated with unit eigenvalue is $\begin{bmatrix} 0.5773 \\ 0.5773 \\ 0.5773 \end{bmatrix}$, which is the "same" as the rotation axis.

We can guess that this eigenvector $\vec{\boldsymbol{v}}$ is always the unit vector of rotation axis. Actually, we can argue that this statement is true. Given that the eigenvalue is 1, we know $\boldsymbol{R}\vec{\boldsymbol{v}} = \vec{\boldsymbol{v}}$. This indicates that any rotation transformation doesn't change the representation of $\vec{\boldsymbol{v}}$, which leaves the only explanation that $\vec{\boldsymbol{v}}$ and rotation axis are collinear vectors.

5. Solution

a) Plot of the path.



The plot shows that the smaller δt is, the smoother the path will appear and the more accurate the path will be compared to circle path.

b) Matlab code of function file: diffDrive.m

```
function [x, y, theta] = diffDrive(s, v, omega, t, delta)
%%% compute the path of differential drive robot given
%%% a constant linear velocity and a constant angular
%%% velocity

    %%% change angles from degree to radian
    s(3) = degtorad(s(3));
    omega = degtorad(omega);

    x=zeros(1,t+1);
    y=zeros(1,t+1);
    theta=zeros(1,t+1);
    %%% x, y, theta index is 1 greater than t
    x_t = s(1);
    y_t = s(2);
    theta_t = s(3);
    %%% initial state of robot
    x(1) = x_t;
    y(1) = y_t;
    theta(1) = theta_t;
    %%% index 1 stands for initial state
```

```
    for i=1:t
        %%% in each time step, all the states change
        %%% base on diffDrive kinematics
        x_t = x_t + v*cos(theta_t)*delta;
        y_t = y_t + v*sin(theta_t)*delta;
        theta_t = theta_t + omega*delta;
        x(i+1) = x_t;
        y(i+1) = y_t;
        theta(i+1) = theta_t;
    end
end
```

c) Matlab code of script file: path_plot.m

```
[x, y, theta] =diffDrive([100,50,45], 1, 2, 10, 9);
figure;
plot(x, y);
hold on

[x, y, theta] =diffDrive([100,50,45], 1, 2, 15, 6);
plot(x, y);
hold on


[x, y, theta] =diffDrive([100,50,45], 1, 2, 30, 3);
plot(x,y);
axis equal

hold off
title('Paths comparison of different \deltat');
xlabel('x')
ylabel('y')
legend('right-most: \deltat=9', 'middle: \deltat=6', ...
    'left-most: \deltat=3', 'Location', 'SouthWest')
```