UNIVERSITÀ DEGLI STUDI DI GENOVA

DIBRIS

DEPARTMENT OF COMPUTER SCIENCE AND TECHNOLOGY,
BIOENGINEERING, ROBOTICS AND SYSTEM ENGINEERING

MODELLING AND CONTROL OF MANIPULATORS

First Assignment

Equivalent representations of orientation matrices

Author:

Triki Karim

Student ID:

s5528602

Professors:

Giovanni Indiveri
Enrico Simetti
Giorgio Cannata

Tutors:

Andrea Tiranti
Francesco Giovinazzo

15 December 2022

Contents

Mathematical expression	Definition	MATLAB expression
$< w >$	World Coordinate Frame	w
${}^{a}_{b}R$	Rotation matrix of frame $< b >$ with respect to frame $< a >$	aRb
${}^{a}_{b}T$	Transformation matrix of frame $< b >$ with respect to frame $< a >$	aTb

Table 1: Nomenclature Table

1 Assignment description

The first assignment of Modelling and Control of Manipulators focuses on the geometric fundamentals and algorithmic tools underlying any robotics application. The concepts of transformation matrix, orientation matrix and the equivalent representations of orientation matrices (Equivalent angle-axis representation, Euler Angles and Quaternions) will be reviewed.

The first assignment is **mandatory** and consists of 4 different exercises. You are asked to:

- Download the .zip file called MOCOM-LAB1 from the Aulaweb page of this course.
- Implement the code to solve the exercises on MATLAB by filling the predefined files called "*main.m*", "*ComputeAngleAxis.m*", "*ComputeInverseAngleAxis.m*", and "*QuatToRot.m*".
- Write a report motivating the answers for each exercise, following the predefind format on this document.

1.1 Exercise 1 - Equivalent Angle-Axis Representation (Exponential representation)

A particularly interesting minimal representation of 3D rotation matrices is the so-called "*angle-axis representation*" or "*exponential representation*". Given two frames $< a >$ and $< b >$, initially coinciding, let's consider an applied geometric unit vector $(\mathbf{v}, O_a) = (\mathbf{v}, O_b)$, passing through the common origin of the two frames, whose initial projection on $< a >$ is the same of that on $< b >$. Then let's consider that frame $< b >$ is purely rotated around **v** of an angle θ, even negative, accordingly with the right-hand rule. We note that the axis-line defined by $(\mathbf{v}, O_a) = (\mathbf{v}, O_b)$ remains common to both the reference systems of the two frames $< a >$ and $< b >$ and we obtain that the orientation matrix constructed in the above way is said to be represented by its equivalent angle-axis representation that admits the following equivalent analytical expression, also known as Rodrigues Formula:

$$\mathbf{R}(^*\mathbf{v}, \theta) = e^{[^*\mathbf{v}\wedge]\theta} = e^{[\rho\wedge]} = \mathbf{I}_{3x3} + [^*\mathbf{v}\wedge]\sin(\theta) + [^*\mathbf{v}\wedge]^2(1 - \cos(\theta))$$

Q1.1 Given two generic frames $< a >$ and $< b >$, given the geometric unit vector $(\mathbf{v}, O_a) = (\mathbf{v}, O_b)$ and the angle θ, implement on MATLAB the Rodrigues formula, computing the rotation matrix ${}^a_bR$ of frame $< b >$ with respect to $< a >$.

Then test it for the following cases and comment the results obtained, including some sketches of the frames configurations:

- **Q1.2** $\mathbf{v} = [1, 0, 0]$ and $\theta = 30°$
- **Q1.3** $\mathbf{v} = [0, 1, 0]$ and $\theta = \pi/4$
- **Q1.4** $\mathbf{v} = [0, 0, 1]$ and $\theta = \pi/2$
- **Q1.5** $\mathbf{v} = [0.408, 0.816, -0.408]$ and $\theta = 0.2449$
- **Q1.6** $\rho = [0, \pi/2, 0];$
- **Q1.7** $\rho = [0.4, -0.3, -0.3];$
- **Q1.8** $\rho = [-\pi/4, -\pi/3, \pi/8];$

1.2 Exercise 2 - Inverse Equivalent Angle-Axis Problem

Given two reference frames $< a >$ and $< b >$, referred to a common world coordinate system $< w >$, their orientation with respect to the world frame $< w >$ is expressed in Figure 12.

Q2.1 Compute the orientation matrix ${}^a_bR$.
Q2.2 Solve the Inverse Equivalent Angle-Axis Problem for the orientation matrix ${}^a_bR$.
Q2.3 Given the following Transformation matrix:

$${}^w_cT = \begin{bmatrix} 0.835959 & -0.283542 & -0.46986 & 0 \\ 0.271321 & 0.957764 & -0.0952472 & -1.23 \\ 0.47703 & -0.0478627 & 0.877583 & 14 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Solve the Inverse Equivalent Angle-Axis Problem for the orientation matrix ${}^c_bR$.
*

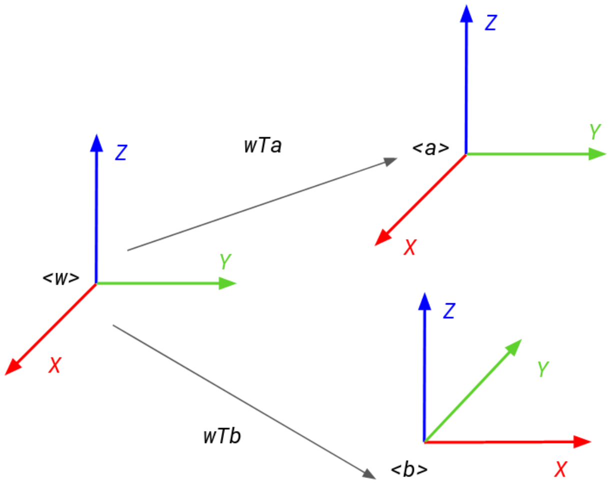

Figure 1: exercise 2 frames

1.3 Exercise 3 - Euler angles (Z-X-Z) vs Tait-Bryan angles (Yaw-Pitch-Roll)

Any orientation matrix can be expressed in terms of three elementary rotations in sequence. These can occur either about the axes of a fixed coordinate system (extrinsic rotations), or about the axes of a rotating coordinate system (intrinsic rotations) initially aligned with the fixed one. Then we can distinguish:

- Proper Euler angles: X-Z-X, Y-Z-Y, ...
- Tait-Bryan angles: Z-Y-X, X-Y-Z, ...

Q3.1 Given two generic frames $< w >$ and $< b >$, define the elementary orientation matrices for frame $< b >$ with respect to frame $< w >$, knowing that:

- $< b >$ is rotated of $30°$ around the z-axis of $< w >$
- $< b >$ is rotated of $45°$ around the y-axis of $< w >$
- $< b >$ is rotated of $15°$ around the x-axis of $< w >$

Q3.2 Compute the equivalent angle-axis representation for each elementary rotation
Q3.3 Compute the z-y-x (yaw,pitch,roll) representation and solve the Inverse Equivalent Angle-Axis Problem for the obtained orientation matrix
Q3.4 Compute the z-x-z representation and solve the Inverse Equivalent Angle-Axis Problem for the obtained orientation matrix

1.4 Exercise 4 - Quaternions

Given the following quaternion: $q = 0.8924 + 0.23912i + 0.36964j + 0.099046k$ expressing how a reference frame $< b >$ is rotated with respect to $< a >$:
Q4.1 Compute the equivalent rotation matrix
Q4.2 Solve the Inverse Equivalent Angle-Axis Problem for the obtained orientation matrix

2 Exercise 1

2.1 Q1.1

In this question we represent Rodrigues Formula On MATLAB that it should compute the rotation matrix

```
function R = ComputeAngleAxis(theta,v)
%Implement here the Rodrigues formula
I = eye(3);
q=[0 -v(3) v(2); v(3) 0 -v(1); -v(2) v(1) 0];
R= I+ q*sin(theta)+ (1-cos(theta))*(q*q);
end
```

Figure 2: Compute Angel Axis

and now we are going to test it in the next questions

2.2 Q1.2

v= [1 0 0]

theta= 30°

theta= pi/6 rad

$$ {}^a_bR = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0.8660 & -0.5 \\ 0 & 0.5 & 0.8660 \end{bmatrix} $$

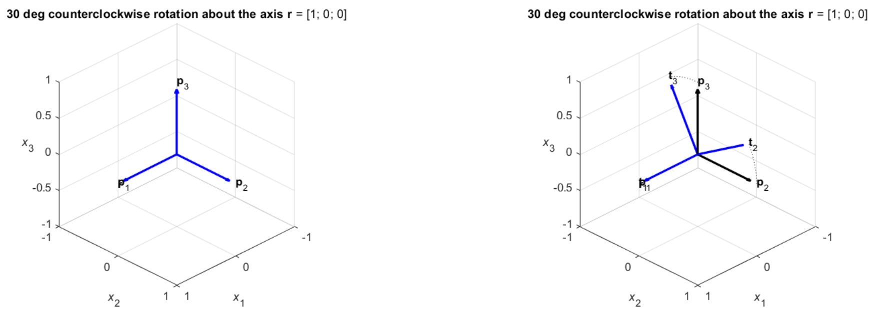

Figure 3: Q1.2

2.3 Q1.3

v= [0 1 0]

theta= pi/4

$$ {}^a_bR = \begin{bmatrix} 0.7071 & 0 & 0.7071 \\ 0 & 1 & 0 \\ -0.7071 & 0 & 0.7071 \end{bmatrix} $$

2.4 Q1.4

v=[0 0 1]

theta= pi/2

$$ {}^a_bR = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} $$

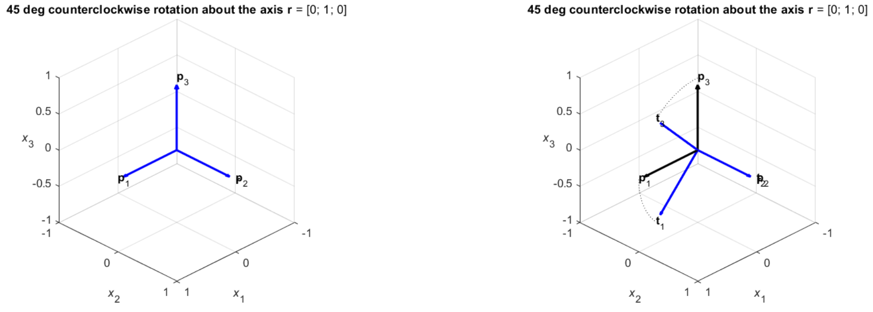

Figure 4: Q1.3

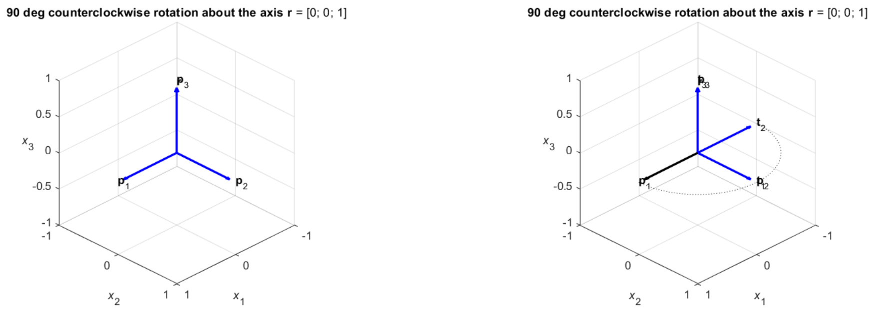

Figure 5: Q1.4

2.5 Q1.5

v=[0.408 0.816 -0.408]
norm vector is not equal to 1 so we should normalize the vector v
v=v/norm(v)
theta= 0.2449

$$
{}^{a}_{b}R = \begin{bmatrix} 0.9751 & 0.1089 & 0.1930 \\ -0.0890 & 0.9901 & -0.1089 \\ -0.2029 & 0.0890 & 0.9751 \end{bmatrix}
$$

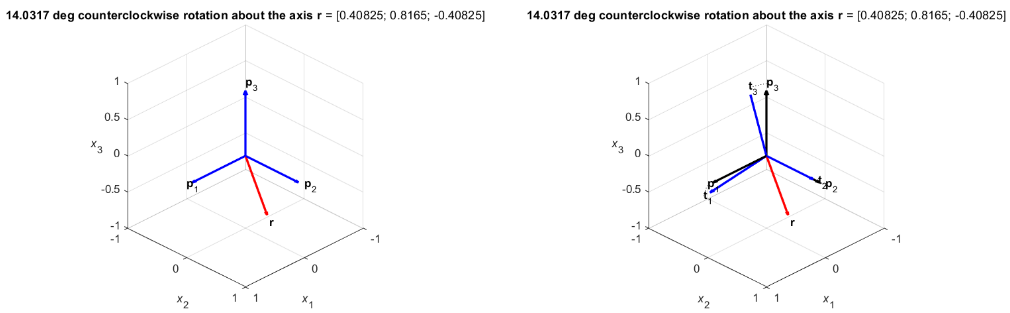

Figure 6: Q1.5

from question Q1.6 to Q1.8 we have the rotation vector so we need to divide it to angel angel axis (theta) and geometric Unit Vector (v)

2.6 Q1.6

p= [0 pi/2 0]
theta=norm(p)
theta= 1.5708
v=p/theta
v=[0 1 0]

$$ {}^{a}_{b}R = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ -1 & 0 & 0 \end{bmatrix} $$

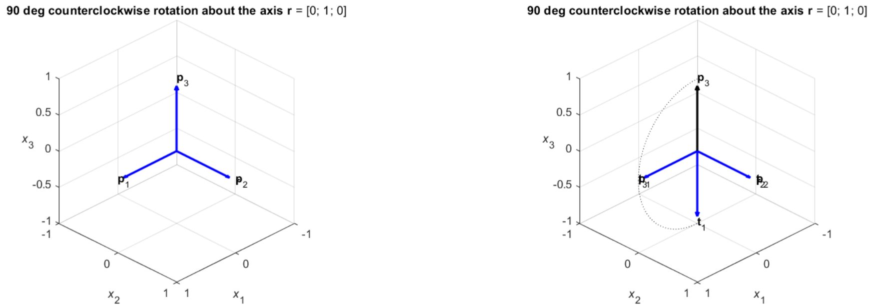

Figure 7: Q1.6

2.7 Q1.7

p= [0.4 -0.3 -0.3]
theta=norm(p)
theta= 0.5831
v=p/theta
v=[0.6860 -0.5145 -0.5145]

$$ {}^{a}_{b}R = \begin{bmatrix} 0.9125 & 0.2250 & -0.3416 \\ -0.3416 & 0.8785 & -0.3340 \\ 0.2250 & 0.4215 & 0.8785 \end{bmatrix} $$

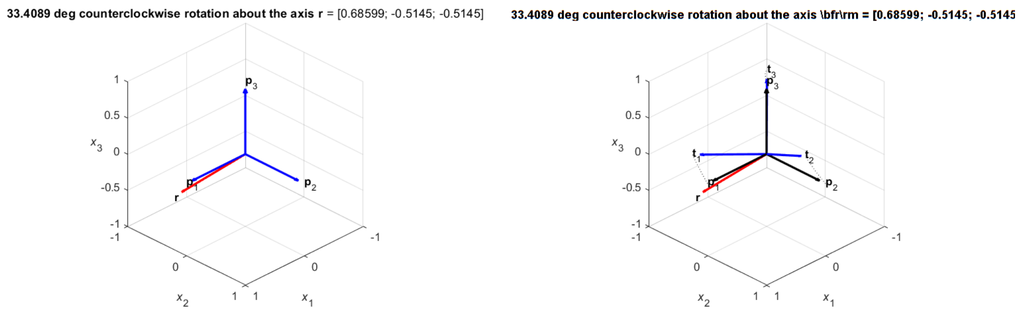

Figure 8: Q1.7

2.8 Q1.8

p= [-pi/4 -pi/3 pi/8]
theta=norm(p)
theta= 1.3666
v=p/theta
v=[-0.5747 -0.7663 0.2873]

$$ {}^{a}_{b}R = \begin{bmatrix} 0.4661 & 0.0697 & -0.8820 \\ 0.6325 & 0.6709 & 0.3872 \\ 0.6187 & -0.7383 & 0.2686 \end{bmatrix} $$

3 Exercise 2

3.1 Q2.1

to Compute the orientation matrix we need the Inverse of Compute Angle Axis
In this situation we have chose the rotation around z-axis and the angle pi/2

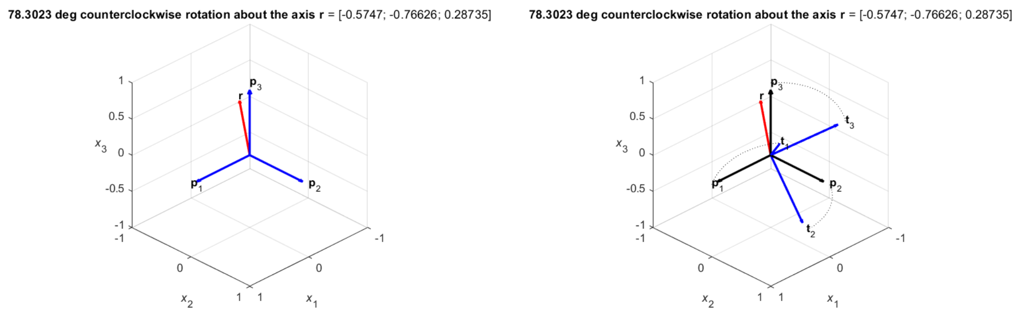

Figure 9: Q1.8

```
function [theta,v] = ComputeInverseAngleAxis(R)
[a, b]=size(R);
  M=(R-R')/2;
   vex=[0.5*(M(3,2)-M(2,3));0.5*(M(1,3)-M(3,1));0.5*(M(2,1)-M(1,2))];
     if a==3 && b==3
         if abs(1-diag(R*R'))<=0.001
           if abs(1-det(R))<=0.001
               theta=acos((trace(R)-1)/2);
               [ev,evl]=eig(R);
               e=abs(1-diag(evl))<=0.001;
               v=ev(:,e);
            else
              error('DETERMINANT OF THE INPUT MATRIX IS 0')
            end
        else
           error('NOT ORTHOGONAL INPUT MATRIX')
          end
    else
       error('WRONG SIZE OF THE INPUT MATRIX')

     end
end

```

Figure 10: Compute Angel Axis Inverse

v=[0 0 1]
theta= pi/2
(a) and (w) are identical and parallel so:

$$ {}^{a}_{w}R = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} $$

$$ {}^{w}_{b}R = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} $$

$$ {}^{a}_{b}R = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} $$

3.2 Q2.2

v=[0 0 1]
theta= 1.5708

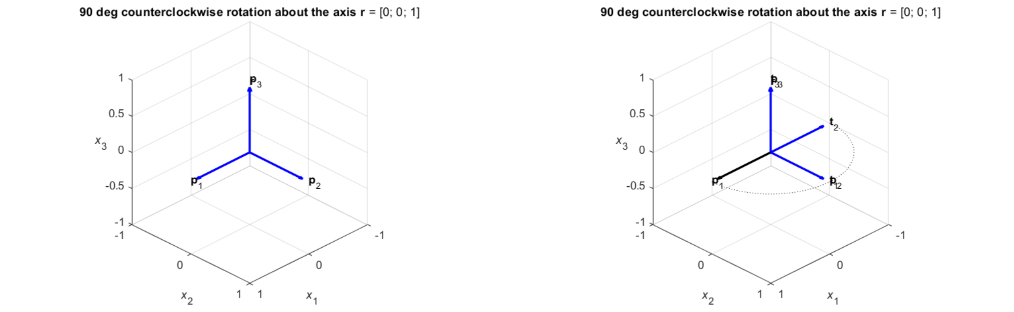

Figure 11: Q2.2

3.3 Q2.3

we have : ${}^{w}_{c}R = \begin{bmatrix} 0.835959 & -0.283542 & -0.469869 \\ 0.271321 & 0.957764 & 0.0952472 \\ 0.47703 & -0.0478627 & 0.877583 \end{bmatrix}$
and
${}^{c}_{b}R$= ${}^{w}_{c}R$'x ${}^{w}_{b}R$
v=[0.2651 0.2931 0.9186]
theta= 1.3529

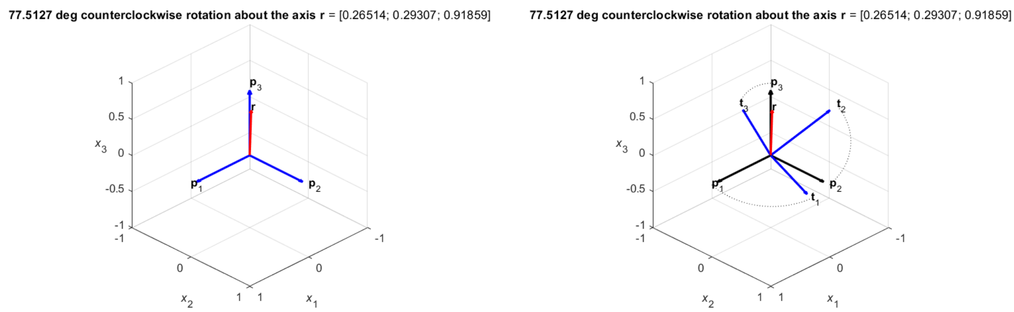

Figure 12: Q2.2

4 Exercise 3

4.1 Q3.1

```
% a
    %rotation matrix from <w> to frame <b> by rotating around z-axes
    wRb_z = [cos(pi/6) -sin(pi/6) 0 ; sin(pi/6) cos(pi/6) 0; 0 0 1];

% b
    %rotation matrix from <w> to frame <b> by rotating around y-axes
     wRb_y = [cos(pi/4) 0 sin(pi/4) ; 0 1 0 ; -sin(pi/4) 0 cos(pi/4)];

% c
    %rotation matrix from <w> to frame <b> by rotating around x-axes
    wRb_x = [1 0 0 ; 0 cos(pi/12) -sin(pi/12) ; 0 sin(pi/12) cos(pi/12)];


disp('es 3.1:');disp(wRb_z);disp(wRb_y);disp(wRb_x);
```

4.2 Q3.2

4.3 a.

v=[0 0 1]
theta=0.5236

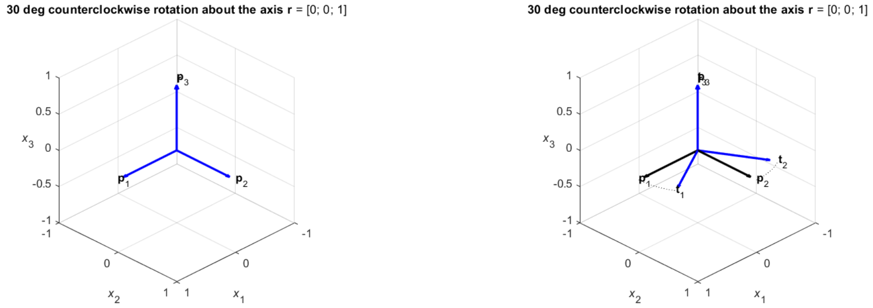

Figure 13: Q3.2a

4.4 b.

v=[0 1 0]
theta= 0.7854

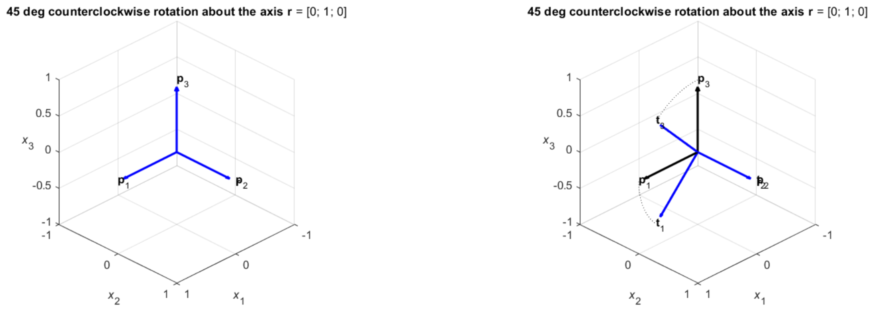

Figure 14: Q3.2b

4.5 c.

v=[1 0 0]
theta=0.2618

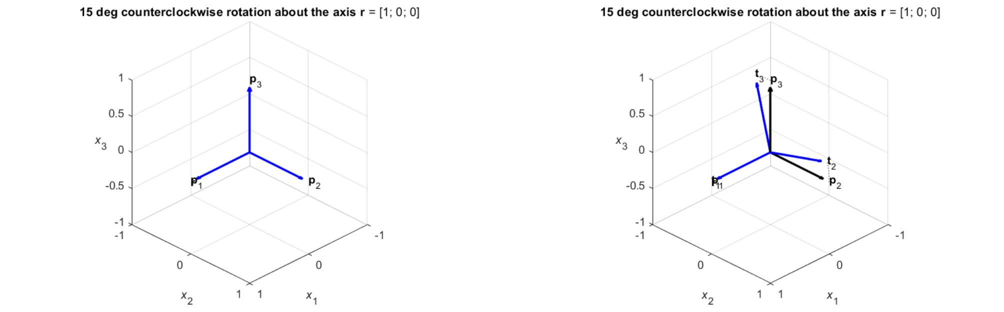

Figure 15: Q3.2c

4.6 Q3.3

v=[0.0415 0.9026 0.4285]
theta= 0.9126

Rzyx 3.3:

0.6124	-0.3245	0.7209
0.3536	0.9280	0.1174
-0.7071	0.1830	0.6830

Figure 16: Matrix of Rzyx

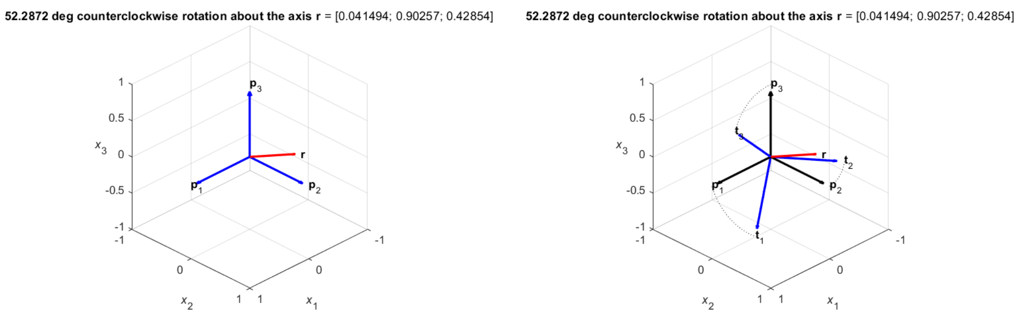

Figure 17: Q3.3

4.7 Q3.4

v=[0.2546 0 0.9670]
theta=1.0765

```
Rzxz 3.4:
    0.5085   -0.8513    0.1294
    0.8513    0.4744   -0.2241
    0.1294    0.2241    0.9659
```

Figure 18: Matrix of Rzxz

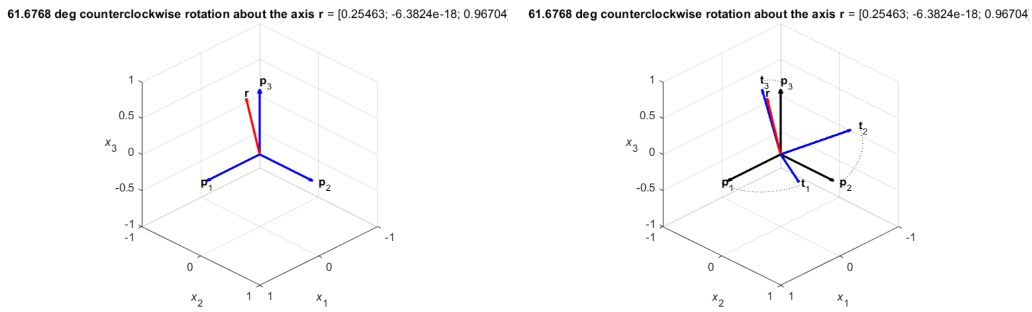

Figure 19: Q3.4

5 Exercise 4

5.1 Q4.1

a = 0.8924;
b = 0.23912;
c = 0.36964;
d = 0.099046;
q=[a b c d];

```
rot matrix es 4.1
    0.7071   -0.0000    0.7071
    0.3536    0.8660   -0.3536
   -0.6124    0.5000    0.6124
```

Figure 20: Rot Matrix 4.1

5.2 Q4.2

```
rot matrix2 es 4.2
     0.7071   -0.0000    0.7071
     0.3536    0.8660   -0.3536
    -0.6124    0.5000    0.6124
```

Figure 21: Rot Matrix 4.2

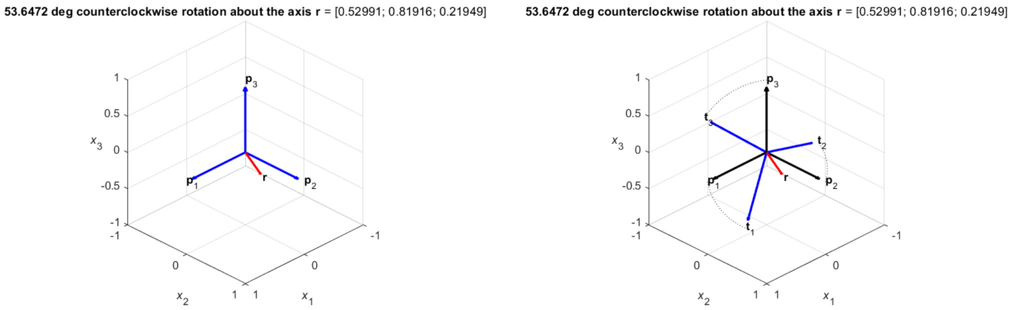

Figure 22: Q4.2