Name: GOVIND AJITH
UID: 116699488

ENPM 662 - 2019

Homework - 1

(1)

(a) To prove $||Rq - Rp|| = ||q - p||$

$\forall q, p \in R^3$

Since R is orthogonal, multiplying R^T on both sides

$$R^T ||Rq - Rp|| = R^T ||q - p||$$

$$||(R^T R)q - R^T(R)p|| = ||R^T(q - p)||$$

Noew: $R^T R = I$ (property of orthogonal matrix)

and $||R^T|| = ||R||$

$\therefore$ $$||q - p|| = ||R(q - p)||$$
$$= ||Rq - Rp||$$

Hence proved

(b) Let the general form of the R matrix be

$$R=\begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix}$$

Let the general format of V be
$V = V_1\hat{i} + V_2\hat{j} + V_3\hat{k}$ and of W be
$W = W_1\hat{i} + W_2\hat{j} + W_3\hat{k}$

Calculating..
$R(V \times W)$

$$=\begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix}\begin{bmatrix} V_2W_3 - W_2V_3 \\ -V_1W_3 + W_1V_3 \\ V_1W_2 - V_2W_1 \end{bmatrix}$$

Calculating the first term we get →

$$r_{11}V_2W_3 - r_{11}W_2V_3 - r_{12}V_1W_3 + r_{12}W_1V_3 + r_{13}V_1W_2 - r_{13}V_2W_1 \longrightarrow L\cdot H\cdot S$$

Now, Calculating $(RV)\times(RW)$ on the $R\cdot H\cdot S$

$$=\begin{bmatrix} r_{11}V_1 + r_{12}V_2 + r_{13}V_3 \\ r_{21}V_1 + r_{22}V_2 + r_{23}V_3 \\ r_{31}V_1 + r_{32}V_2 + r_{33}V_3 \end{bmatrix}\begin{bmatrix} r_{11}W_1 + r_{12}W_2 + r_{13}W_3 \\ r_{21}W_1 + r_{22}W_2 + r_{23}W_3 \\ r_{31}W_1 + r_{32}W_2 + r_{33}W_3 \end{bmatrix}$$

Since we are only comparing the first terms...

$$(r_{21}V_1 + r_{22}V_2 + r_{23}V_3)(r_{31}W_1 + r_{32}W_2 + r_{33}W_3)$$

$$-(r_{21}W_1 + r_{22}W_2 + r_{23}W_3)(r_{31}V_1 + r_{32}V_2 + r_{33}V_3)$$

$$= r_{21}V_1 r_{31}W_1 + r_{21}V_1 r_{32}W_2 + r_{21}V_1 r_{33}W_3$$

$$+ r_{22}V_2 r_{31}W_1 + r_{22}V_2 r_{32}W_2 + r_{22}V_2 r_{33}W_3$$

$$+ r_{23}V_3 r_{31}W_1 + r_{23}V_3 r_{32}W_2 + r_{23}V_3 r_{33}W_3$$

$$- r_{21}W_1 r_{31}V_1 - r_{21}W_1 r_{32}V_2 - r_{21}W_1 r_{33}V_3$$

$$- r_{22}W_2 r_{31}V_1 - r_{22}W_2 r_{32}V_2 - r_{22}W_2 r_{33}V_3$$

$$- r_{23}W_3 r_{31}V_1 - r_{23}W_3 r_{32}V_2 - r_{23}W_3 r_{33}V_3$$

After cancelling out the terms and accounting for the fact that each element of a real orthogonal matrix is equal to its cofactor we can write the remainder of the terms as follows:

$$= V_1W_2\underbrace{[r_{21}r_{32} - r_{22}r_{31}]}_{\text{cofactor of } r_{13}} + (-V_1W_3)\underbrace{[r_{21}r_{33} - r_{23}r_{31}]}_{\text{cofactor of } r_{12}}$$

$$+ (-V_2W_1)\underbrace{[r_{22}r_{31} - r_{21}r_{32}]}_{\text{cofactor of } r_{13}} + V_2W_3\underbrace{[r_{22}r_{33} - r_{23}r_{32}]}_{\text{cofactor of } r_{11}}$$

$$-(-w_1 v_3)[\underbrace{r_{21} r_{33} - r_{23} r_{31}}_{\text{Cofactor of } r_{12}}] - w_2 v_3 [\underbrace{r_{22} r_{33} - r_{33} r_{32}}_{\text{Cofactor of } r_{11}}]$$

$$= r_{11} v_2 w_3 - r_{11} w_2 v_3 - r_{12} v_1 w_3 + r_{12} v_3 w_1 + r_{13} v_1 w_2$$

$$- r_{13} v_2 w_1 = \underline{\underline{R \cdot H \cdot S}} = \underline{\underline{L \cdot H \cdot S}}$$

Hence proved that

$$\boxed{R(v \times w) = (Rv) \times (Rw)}$$

(2) The Rotation matrices can be expressed in terms of quaternion

$R_{x,\phi} = \cos\frac{\phi}{2} + \sin\frac{\phi}{2}\,\hat{i}$

$R_{z,\theta} = \cos\frac{\theta}{2} + \sin\frac{\theta}{2}\,\hat{k}$

$R_{y,\psi} = \cos\psi + \sin\frac{\psi}{2}\,\hat{j}$

Now, $(R_{x,\phi})(R_{z,\theta})(R_{y,\phi})$

$= \left(\cos\frac{\phi}{2} + \sin\frac{\phi}{2}\,\hat{i}\right)\left(\cos\frac{\theta}{2} + \sin\frac{\theta}{2}\,\hat{k}\right)\left(\cos\frac{\psi}{2} + \sin\frac{\psi}{2}\,\hat{j}\right)$

$= \left(\cos\frac{\theta}{2}\cos\frac{\phi}{2} + \cos\frac{\phi}{2}\sin\frac{\theta}{2}\,\hat{k} + \cos\frac{\theta}{2}\sin\frac{\phi}{2}\,\hat{i} + \sin\frac{\theta}{2}\sin\frac{\phi}{2}(-\hat{j})\right)\left(\cos\frac{\psi}{2} + \sin\frac{\psi}{2}\,\hat{j}\right)$

Now onwards writing $\cos\frac{\theta}{2}$, $\cos\frac{\phi}{2}$, $\cos\frac{\psi}{2}$, $\sin\frac{\theta}{2}$, $\sin\frac{\phi}{2}$ and $\sin\frac{\psi}{2}$ as $C\theta, C\phi, C\psi, S\theta, S\phi, S\psi$ respectively.

$C\theta C\phi C\psi + C\theta C\phi S\psi\,\hat{j} + C\phi S\theta C\psi\,\hat{k} + C\phi S\theta S\psi(-\hat{i})$
$+ C\theta S\phi C\psi\,\hat{i} + C\theta S\phi S\psi\,\hat{k} - S\theta S\phi C\psi\,\hat{j} + S\theta S\phi S\psi$

$= c\theta c\phi c\psi + s\theta s\phi s\psi + (c\theta s\phi c\psi - s\theta c\phi s\psi)\hat{i}$
$+ (c\theta c\phi s\psi - s\theta s\phi c\psi)\hat{j} + (s\theta c\phi c\psi + c\theta s\phi s\psi)\hat{k}$

$A = \cos\frac{\theta}{2}\cos\frac{\phi}{2}\cos\frac{\psi}{2} + \sin\frac{\theta}{2}\sin\frac{\phi}{2}\sin\frac{\psi}{2}$

$u_x = \cos\frac{\theta}{2}\sin\frac{\phi}{2}\cos\frac{\psi}{2} - \sin\frac{\theta}{2}\cos\frac{\phi}{2}\sin\frac{\psi}{2}$

$u_y = \cos\frac{\theta}{2}\cos\frac{\phi}{2}\sin\frac{\psi}{2} - \sin\frac{\theta}{2}\sin\frac{\phi}{2}\cos\frac{\psi}{2}$

$u_z = \sin\frac{\theta}{2}\cos\frac{\phi}{2}\cos\frac{\psi}{2} + \cos\frac{\theta}{2}\sin\frac{\phi}{2}\sin\frac{\psi}{2}$

As calculated $\sqrt{A^2 + u_x^2 + u_y^2 + u_z^2}$, should be given a value of 1 in order to signify a unit quaternion

To verify

$A^2 = c^2\theta c^2\phi c^2\psi + s^2\theta s^2\phi s^2\psi + 2c\theta c\phi c\psi s\theta s\phi s\psi$

$u_x^2 = c^2\theta s^2\phi c^2\psi + s^2\theta c^2\phi s^2\psi - 2c\theta c\phi c\psi s\theta s\phi s\psi$

$u_y^2 = c^2\theta c^2\phi s^2\psi + s^2\theta s^2\phi c^2\psi - 2c\theta c\phi c\psi s\theta s\phi s\psi$

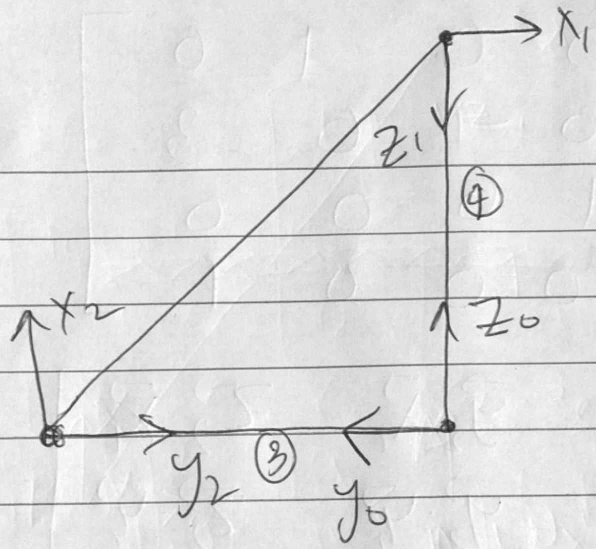
$u_z^2 = s^2\theta c^2\phi c^2\psi + c^2\theta s^2\phi s^2\psi + 2c\theta c\phi c\psi s\theta s\phi s\psi$

Now adding A^2, u_x^2, u_y^2 and u_z^2 $\longrightarrow$

$$= c^2\theta c^2\phi c^2\psi + c^2\theta c^2\phi s^2\psi + c^2\theta s^2\phi c^2\psi$$
$$+ c^2\theta s^2\phi s^2\psi + s^2\theta s^2\phi s^2\psi + s^2\theta s^2\phi c^2\psi$$
$$+ s^2\theta c^2\phi s^2\psi + s^2\theta c^2\phi c^2\psi$$
$$+ 4c\theta c\phi c\psi s\theta s\phi s\psi - 4c\theta c\phi c\psi s\theta s\phi s\psi$$
$$= c^2\theta c^2\phi (c^2\psi + s^2\psi) + c^2\theta s^2\phi (c^2\psi + s^2\psi)$$
$$+ s^2\theta s^2\phi (s^2\psi + c^2\psi) + s^2\theta c^2\phi (s^2\psi + c^2\psi)$$
$$= c^2\theta c^2\phi + c^2\theta s^2\phi + s^2\theta s^2\phi + s^2\theta c^2\phi$$
$$= c^2\theta (c^2\phi + s^2\phi) + s^2\theta (s^2\phi + c^2\phi)$$
$$= c^2\theta + s^2\theta = \underline{\underline{1}}$$

<u>Hence proved</u>

(3)



The rotation matrix between Frame '0' and Frame '1' will be of the form:

$$R_1^0 = \begin{bmatrix} \vec{x}_1 \cdot \vec{x}_0 & \vec{y}_1 \cdot \vec{x}_0 & \vec{z}_1 \cdot \vec{x}_0 \\ \vec{x}_1 \cdot \vec{y}_0 & \vec{y}_1 \cdot \vec{y}_0 & \vec{z}_1 \cdot \vec{y}_0 \\ \vec{x}_1 \cdot \vec{z}_0 & \vec{y}_1 \cdot \vec{z}_0 & \vec{z}_1 \cdot \vec{z}_0 \end{bmatrix}$$

Hence this along with a translation of 4 along the 'Z' axis will be represented as:

$$H_1^0 = \begin{bmatrix} 0 & -1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & -1 & 4 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Similarly $R_2^0 = \begin{bmatrix} \vec{x}_2 \cdot \vec{x}_0 & \vec{y}_2 \cdot \vec{x}_0 & \vec{z}_2 \cdot \vec{x}_0 \\ \vec{x}_2 \cdot \vec{y}_0 & \vec{y}_2 \cdot \vec{y}_0 & \vec{z}_2 \cdot \vec{y}_0 \\ \vec{x}_2 \cdot \vec{z}_0 & \vec{y}_2 \cdot \vec{z}_0 & \vec{z}_2 \cdot \vec{z}_0 \end{bmatrix}$

Hence this along with a translation of 3 along the Y axis can be represented as

$$H_2^0 = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & -1 & 0 & 3 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Now,

$$R_2^1 = \begin{bmatrix} \vec{x}_2 \cdot \vec{x}_1 & \vec{y}_2 \cdot \vec{x}_1 & \vec{z}_2 \cdot \vec{x}_1 \\ \vec{x}_2 \cdot \vec{y}_1 & \vec{y}_2 \cdot \vec{y}_1 & \vec{z}_2 \cdot \vec{y}_1 \\ \vec{x}_2 \cdot \vec{z}_1 & \vec{y}_2 \cdot \vec{z}_1 & \vec{z}_2 \cdot \vec{z}_1 \end{bmatrix}$$

This alongwith an translation of -3 along X and 4 units along Z gives us a homogeneous transformation represented as H_2^1

$$H_2^1 = \begin{bmatrix} 0 & 1 & 0 & -3 \\ 0 & 0 & -1 & 0 \\ -1 & 0 & 0 & 4 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Now: $H_1^0 \times H_2^1$

$$= \begin{bmatrix} 0 & -1 & 0 & 0 \\ -1 & 0 & 0 & 0 \\ 0 & 0 & -1 & 4 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 & 1 & 0 & -3 \\ 0 & 0 & -1 & 0 \\ -1 & 0 & 0 & 4 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$= H_2^0$. <u>Hence proved.</u> [Solution provided in MATLAB as well]

```
%%verification of the answer obtained for question 3
clc
clear all%clearing all the previous outputs and stored variables
%%first Homogeneous matrix
h0_1=[0 -1 0 0;-1 0 0 0;0 0 -1 4;0 0 0 1]
%%second homogeneous matrix
h0_2=[0 0 1 0;0 -1 0 3;1 0 0 0;0 0 0 1]
%%third homogeneous matrix
h1_2=[0 1 0 -3;0 0 -1 0;-1 0 0 4;0 0 0 1]
fprintf('now multiplying the matrices on the RHS: ');
rhs=h0_1*h1_2%the product of the multiplication on the R.H.S
fprintf('now multiplying the matrices on the LHS: ');
lhs=h0_2%the product of the multiplication on the L.H.S
fprintf('the boolean representing the equality can be seen below ');
lhs==rhs%Verifying Equality
```

```
h0_1 =

     0    -1     0     0
    -1     0     0     0
     0     0    -1     4
     0     0     0     1


h0_2 =

     0     0     1     0
     0    -1     0     3
     1     0     0     0
     0     0     0     1


h1_2 =

     0     1     0    -3
     0     0    -1     0
    -1     0     0     4
     0     0     0     1

now multiplying the matrices on the RHS:
rhs =

     0     0     1     0
     0    -1     0     3
     1     0     0     0
     0     0     0     1

now multiplying the matrices on the LHS:
lhs =

     0     0     1     0
```

```
     0    -1     0     3
     1     0     0     0
     0     0     0     1

the boolean representing the equality can be seen below
ans =

  4×4 logical array

   1   1   1   1
   1   1   1   1
   1   1   1   1
   1   1   1   1
```

Published with MATLAB® R2018a

```
%Answer to Question 4
clc
clear all%clearing all previous outputs and variables

syms a d cos(theta) sin(theta) cos(alpha) sin(alpha) %Defining all the
 variables through syms
r_z_theta=[cos(theta) -sin(theta) 0 0;sin(theta) cos(theta) 0 0;0 0 1
 0;0 0 0 1];%defining rotation through theta around the z axis
trform_z_d=[1 0 0 0;0 1 0 0;0 0 1 d;0 0 0 1];%translation by d along
 the z axis
trform_x_a=[1 0 0 a;0 1 0 0;0 0 1 0;0 0 0 1];%translation by a along
 the x axis
r_x_alpha=[1 0 0 0;0 cos(alpha) -sin(alpha) 0;0 sin(alpha) cos(alpha)
 0;0 0 0 1];%defining rotation through alpha around the x axis

H=r_z_theta*trform_z_d*trform_x_a*r_x_alpha %the combination of
 rotation and transformation matrix as
%described in the problem

%below given are all the other combinations of the Homogeneous
 matrices
H_1= r_z_theta*trform_z_d*trform_x_a*r_x_alpha
H_2= r_z_theta*trform_z_d*r_x_alpha*trform_x_a
H_3= r_z_theta*trform_x_a*trform_z_d*r_x_alpha
H_4= r_z_theta*trform_x_a*r_x_alpha*trform_z_d
H_5= r_z_theta*r_x_alpha*trform_z_d*trform_x_a
H_6=r_z_theta*r_x_alpha*trform_x_a*trform_z_d
H_7= trform_z_d*r_z_theta*trform_x_a*r_x_alpha
H_8= trform_z_d*r_z_theta*r_x_alpha*trform_x_a
H_9= trform_z_d*trform_x_a*r_z_theta*r_x_alpha
H_10= trform_z_d*trform_x_a*r_x_alpha*r_z_theta
H_11= trform_z_d*r_x_alpha*r_z_theta*trform_z_d
H_12= trform_z_d*r_x_alpha*trform_z_d*r_z_theta
H_13=trform_x_a*r_z_theta*trform_z_d*r_x_alpha
H_14=trform_x_a*r_z_theta*r_x_alpha*trform_z_d
H_15=trform_x_a*trform_z_d*r_z_theta*r_x_alpha
H_16=trform_x_a*trform_z_d*r_x_alpha*r_z_theta
H_17=trform_x_a*r_x_alpha*r_z_theta*trform_z_d
H_18=trform_x_a*r_x_alpha*trform_z_d*r_z_theta
H_19=r_x_alpha*r_z_theta*trform_z_d*trform_x_a
H_20=r_x_alpha*r_z_theta*trform_x_a*trform_z_d
H_21=r_x_alpha*trform_z_d*r_z_theta*trform_x_a
H_22=r_x_alpha*trform_z_d*trform_x_a*r_z_theta
H_23=r_x_alpha*trform_x_a*r_z_theta*trform_z_d
H_24=r_x_alpha*trform_x_a*trform_z_d*r_z_theta

% checking the equality between H and various other combinations
fprintf('Comparing H and H_1:
 r_z_theta*trform_z_d*trform_x_a*r_x_alpha')
isequal(H,H_1)
fprintf('Comparing H and
 H_2=r_z_theta*trform_z_d*r_x_alpha*trform_x_a')
```

```
isequal(H,H_2)
fprintf('Comparing H and H_3=
 r_z_theta*trform_x_a*trform_z_d*r_x_alpha')
isequal(H,H_3)
fprintf('Comparing H and H_4=
 r_z_theta*trform_x_a*r_x_alpha*trform_z_d')
isequal(H,H_4)
fprintf('Comparing H and H_5=
 r_z_theta*r_x_alpha*trform_z_d*trform_x_a')
isequal(H,H_5)
fprintf('Comparing H and
 H_6=r_z_theta*r_x_alpha*trform_x_a*trform_z_d')
isequal(H,H_6)
fprintf('Comparing H and H_7=
 trform_z_d*r_z_theta*trform_x_a*r_x_alpha')
isequal(H,H_7)
fprintf('Comparing H and H_8=
 trform_z_d*r_z_theta*r_x_alpha*trform_x_a')
isequal(H,H_8)
fprintf('Comparing H and H_9=
 trform_z_d*trform_x_a*r_z_theta*r_x_alpha')
isequal(H,H_9)
fprintf('Comparing H and H_10=
 trform_z_d*trform_x_a*r_x_alpha*r_z_theta')
isequal(H,H_10)
fprintf('Comparing H and H_11=
 trform_z_d*r_x_alpha*r_z_theta*trform_z_d')
isequal(H,H_11)
fprintf('Comparing H and H_12=
 trform_z_d*r_x_alpha*trform_z_d*r_z_theta')
isequal(H,H_12)
fprintf('Comparing H and
 H_13=trform_x_a*r_z_theta*trform_z_d*r_x_alpha')
isequal(H,H_13)
fprintf('Comparing H and
 H_14=trform_x_a*r_z_theta*r_x_alpha*trform_z_d')
isequal(H,H_14)
fprintf('Comparing H and
 H_15=trform_x_a*trform_z_d*r_z_theta*r_x_alpha')
isequal(H,H_15)
fprintf('Comparing H and
 H_16=trform_x_a*trform_z_d*r_x_alpha*r_z_theta')
isequal(H,H_16)
fprintf('Comparing H and
 H_17=trform_x_a*r_x_alpha*r_z_theta*trform_z_d')
isequal(H,H_17)
fprintf('Comparing H and
 H_18=trform_x_a*r_x_alpha*trform_z_d*r_z_theta')
isequal(H,H_18)
fprintf('Comparing H and
 H_19=r_x_alpha*r_z_theta*trform_z_d*trform_x_a')
isequal(H,H_19)
fprintf('Comparing H and
 H_20=r_x_alpha*r_z_theta*trform_x_a*trform_z_d')
```

```
isequal(H,H_20)
fprintf('Comparing H and
 H_21=r_x_alpha*trform_z_d*r_z_theta*trform_x_a')
isequal(H,H_21)
fprintf('Comparing H and
 H_22=r_x_alpha*trform_z_d*trform_x_a*r_z_theta')
isequal(H,H_22)
fprintf('Comparing H and
 H_23=r_x_alpha*trform_x_a*r_z_theta*trform_z_d')
isequal(H,H_23)
fprintf('Comparing H and
 H_24=r_x_alpha*trform_x_a*trform_z_d*r_z_theta')
isequal(H,H_24)
```

```
H =

[ cos(theta), -cos(alpha)*sin(theta),  sin(alpha)*sin(theta),
 a*cos(theta)]
[ sin(theta),  cos(alpha)*cos(theta), -sin(alpha)*cos(theta),
 a*sin(theta)]
[          0,             sin(alpha),             cos(alpha),
    d]
[          0,                      0,                      0,
    1]


H_1 =

[ cos(theta), -cos(alpha)*sin(theta),  sin(alpha)*sin(theta),
 a*cos(theta)]
[ sin(theta),  cos(alpha)*cos(theta), -sin(alpha)*cos(theta),
 a*sin(theta)]
[          0,             sin(alpha),             cos(alpha),
    d]
[          0,                      0,                      0,
    1]


H_2 =

[ cos(theta), -cos(alpha)*sin(theta),  sin(alpha)*sin(theta),
 a*cos(theta)]
[ sin(theta),  cos(alpha)*cos(theta), -sin(alpha)*cos(theta),
 a*sin(theta)]
[          0,             sin(alpha),             cos(alpha),
    d]
[          0,                      0,                      0,
    1]


H_3 =
```

```
[ cos(theta), -cos(alpha)*sin(theta),  sin(alpha)*sin(theta),
 a*cos(theta)]
[ sin(theta),  cos(alpha)*cos(theta), -sin(alpha)*cos(theta),
 a*sin(theta)]
[          0,             sin(alpha),             cos(alpha),
    d]
[          0,                      0,                      0,
    1]


H_4 =

[ cos(theta), -cos(alpha)*sin(theta),  sin(alpha)*sin(theta),
 a*cos(theta) + d*sin(alpha)*sin(theta)]
[ sin(theta),  cos(alpha)*cos(theta), -sin(alpha)*cos(theta),
 a*sin(theta) - d*sin(alpha)*cos(theta)]
[          0,             sin(alpha),             cos(alpha),
                  d*cos(alpha)]
[          0,                      0,                      0,
                             1]


H_5 =

[ cos(theta), -cos(alpha)*sin(theta),  sin(alpha)*sin(theta),
 a*cos(theta) + d*sin(alpha)*sin(theta)]
[ sin(theta),  cos(alpha)*cos(theta), -sin(alpha)*cos(theta),
 a*sin(theta) - d*sin(alpha)*cos(theta)]
[          0,             sin(alpha),             cos(alpha),
                  d*cos(alpha)]
[          0,                      0,                      0,
                             1]


H_6 =

[ cos(theta), -cos(alpha)*sin(theta),  sin(alpha)*sin(theta),
 a*cos(theta) + d*sin(alpha)*sin(theta)]
[ sin(theta),  cos(alpha)*cos(theta), -sin(alpha)*cos(theta),
 a*sin(theta) - d*sin(alpha)*cos(theta)]
[          0,             sin(alpha),             cos(alpha),
                  d*cos(alpha)]
[          0,                      0,                      0,
                             1]


H_7 =

[ cos(theta), -cos(alpha)*sin(theta),  sin(alpha)*sin(theta),
 a*cos(theta)]
[ sin(theta),  cos(alpha)*cos(theta), -sin(alpha)*cos(theta),
 a*sin(theta)]
[          0,             sin(alpha),             cos(alpha),
    d]
```

```
[           0,                        0,                        0,
    1]


H_8 =

[ cos(theta), -cos(alpha)*sin(theta),  sin(alpha)*sin(theta),
 a*cos(theta)]
[ sin(theta),  cos(alpha)*cos(theta), -sin(alpha)*cos(theta),
 a*sin(theta)]
[           0,             sin(alpha),             cos(alpha),
    d]
[           0,                        0,                        0,
    1]


H_9 =

[ cos(theta), -cos(alpha)*sin(theta),  sin(alpha)*sin(theta), a]
[ sin(theta),  cos(alpha)*cos(theta), -sin(alpha)*cos(theta), 0]
[           0,             sin(alpha),             cos(alpha), d]
[           0,                        0,                        0, 1]


H_10 =

[             cos(theta),            -sin(theta),           0, a]
[ cos(alpha)*sin(theta), cos(alpha)*cos(theta), -sin(alpha), 0]
[ sin(alpha)*sin(theta), sin(alpha)*cos(theta),  cos(alpha), d]
[                       0,                        0,           0, 1]


H_11 =

[             cos(theta),            -sin(theta),           0,
      0]
[ cos(alpha)*sin(theta), cos(alpha)*cos(theta), -sin(alpha),    -
d*sin(alpha)]
[ sin(alpha)*sin(theta), sin(alpha)*cos(theta),  cos(alpha), d +
 d*cos(alpha)]
[                       0,                        0,           0,
      1]


H_12 =

[             cos(theta),            -sin(theta),           0,
      0]
[ cos(alpha)*sin(theta), cos(alpha)*cos(theta), -sin(alpha),    -
d*sin(alpha)]
[ sin(alpha)*sin(theta), sin(alpha)*cos(theta),  cos(alpha), d +
 d*cos(alpha)]
[                       0,                        0,           0,
      1]
```

```
H_13 =

[ cos(theta), -cos(alpha)*sin(theta),  sin(alpha)*sin(theta), a]
[ sin(theta),  cos(alpha)*cos(theta), -sin(alpha)*cos(theta), 0]
[          0,             sin(alpha),             cos(alpha), d]
[          0,                      0,                      0, 1]


H_14 =

[ cos(theta), -cos(alpha)*sin(theta),  sin(alpha)*sin(theta), a +
 d*sin(alpha)*sin(theta)]
[ sin(theta),  cos(alpha)*cos(theta), -sin(alpha)*cos(theta),    -
d*sin(alpha)*cos(theta)]
[          0,             sin(alpha),             cos(alpha),
       d*cos(alpha)]
[          0,                      0,                      0,
                  1]


H_15 =

[ cos(theta), -cos(alpha)*sin(theta),  sin(alpha)*sin(theta), a]
[ sin(theta),  cos(alpha)*cos(theta), -sin(alpha)*cos(theta), 0]
[          0,             sin(alpha),             cos(alpha), d]
[          0,                      0,                      0, 1]


H_16 =

[            cos(theta),            -sin(theta),           0, a]
[ cos(alpha)*sin(theta), cos(alpha)*cos(theta), -sin(alpha), 0]
[ sin(alpha)*sin(theta), sin(alpha)*cos(theta),  cos(alpha), d]
[                     0,                     0,           0, 1]


H_17 =

[            cos(theta),            -sin(theta),           0,
    a]
[ cos(alpha)*sin(theta), cos(alpha)*cos(theta), -sin(alpha), -
d*sin(alpha)]
[ sin(alpha)*sin(theta), sin(alpha)*cos(theta),  cos(alpha),
 d*cos(alpha)]
[                     0,                     0,           0,
    1]


H_18 =

[            cos(theta),            -sin(theta),           0,
    a]
```

```
[ cos(alpha)*sin(theta), cos(alpha)*cos(theta), -sin(alpha), -
d*sin(alpha)]
[ sin(alpha)*sin(theta), sin(alpha)*cos(theta),  cos(alpha),
 d*cos(alpha)]
[                     0,                     0,           0,
    1]


H_19 =

[            cos(theta),            -sin(theta),           0,
                 a*cos(theta)]
[ cos(alpha)*sin(theta), cos(alpha)*cos(theta), -sin(alpha),
 a*cos(alpha)*sin(theta) - d*sin(alpha)]
[ sin(alpha)*sin(theta), sin(alpha)*cos(theta),  cos(alpha),
 d*cos(alpha) + a*sin(alpha)*sin(theta)]
[                     0,                     0,           0,
                            1]


H_20 =

[            cos(theta),            -sin(theta),           0,
                 a*cos(theta)]
[ cos(alpha)*sin(theta), cos(alpha)*cos(theta), -sin(alpha),
 a*cos(alpha)*sin(theta) - d*sin(alpha)]
[ sin(alpha)*sin(theta), sin(alpha)*cos(theta),  cos(alpha),
 d*cos(alpha) + a*sin(alpha)*sin(theta)]
[                     0,                     0,           0,
                            1]


H_21 =

[            cos(theta),            -sin(theta),           0,
                 a*cos(theta)]
[ cos(alpha)*sin(theta), cos(alpha)*cos(theta), -sin(alpha),
 a*cos(alpha)*sin(theta) - d*sin(alpha)]
[ sin(alpha)*sin(theta), sin(alpha)*cos(theta),  cos(alpha),
 d*cos(alpha) + a*sin(alpha)*sin(theta)]
[                     0,                     0,           0,
                            1]


H_22 =

[            cos(theta),            -sin(theta),           0,
    a]
[ cos(alpha)*sin(theta), cos(alpha)*cos(theta), -sin(alpha), -
d*sin(alpha)]
[ sin(alpha)*sin(theta), sin(alpha)*cos(theta),  cos(alpha),
 d*cos(alpha)]
[                     0,                     0,           0,
    1]
```

```
H_23 =

[            cos(theta),           -sin(theta),           0,
    a]
[ cos(alpha)*sin(theta), cos(alpha)*cos(theta), -sin(alpha), -
d*sin(alpha)]
[ sin(alpha)*sin(theta), sin(alpha)*cos(theta),  cos(alpha),
 d*cos(alpha)]
[                     0,                     0,           0,
    1]


H_24 =

[            cos(theta),           -sin(theta),           0,
    a]
[ cos(alpha)*sin(theta), cos(alpha)*cos(theta), -sin(alpha), -
d*sin(alpha)]
[ sin(alpha)*sin(theta), sin(alpha)*cos(theta),  cos(alpha),
 d*cos(alpha)]
[                     0,                     0,           0,
    1]

Comparing H and H_1: r_z_theta*trform_z_d*trform_x_a*r_x_alpha
ans =

  logical

   1

Comparing H and H_2=r_z_theta*trform_z_d*r_x_alpha*trform_x_a
ans =

  logical

   1

Comparing H and H_3= r_z_theta*trform_x_a*trform_z_d*r_x_alpha
ans =

  logical

   1

Comparing H and H_4= r_z_theta*trform_x_a*r_x_alpha*trform_z_d
ans =

  logical

   0

Comparing H and H_5= r_z_theta*r_x_alpha*trform_z_d*trform_x_a
```

```
ans =

  logical

   0

Comparing H and H_6=r_z_theta*r_x_alpha*trform_x_a*trform_z_d
ans =

  logical

   0

Comparing H and H_7= trform_z_d*r_z_theta*trform_x_a*r_x_alpha
ans =

  logical

   1

Comparing H and H_8= trform_z_d*r_z_theta*r_x_alpha*trform_x_a
ans =

  logical

   1

Comparing H and H_9= trform_z_d*trform_x_a*r_z_theta*r_x_alpha
ans =

  logical

   0

Comparing H and H_10= trform_z_d*trform_x_a*r_x_alpha*r_z_theta
ans =

  logical

   0

Comparing H and H_11= trform_z_d*r_x_alpha*r_z_theta*trform_z_d
ans =

  logical

   0

Comparing H and H_12= trform_z_d*r_x_alpha*trform_z_d*r_z_theta
ans =

  logical

   0
```

```
Comparing H and H_13=trform_x_a*r_z_theta*trform_z_d*r_x_alpha
ans =

  logical

   0

Comparing H and H_14=trform_x_a*r_z_theta*r_x_alpha*trform_z_d
ans =

  logical

   0

Comparing H and H_15=trform_x_a*trform_z_d*r_z_theta*r_x_alpha
ans =

  logical

   0

Comparing H and H_16=trform_x_a*trform_z_d*r_x_alpha*r_z_theta
ans =

  logical

   0

Comparing H and H_17=trform_x_a*r_x_alpha*r_z_theta*trform_z_d
ans =

  logical

   0

Comparing H and H_18=trform_x_a*r_x_alpha*trform_z_d*r_z_theta
ans =

  logical

   0

Comparing H and H_19=r_x_alpha*r_z_theta*trform_z_d*trform_x_a
ans =

  logical

   0

Comparing H and H_20=r_x_alpha*r_z_theta*trform_x_a*trform_z_d
ans =

  logical
```

```
   0

Comparing H and H_21=r_x_alpha*trform_z_d*r_z_theta*trform_x_a
ans =

  logical

   0

Comparing H and H_22=r_x_alpha*trform_z_d*trform_x_a*r_z_theta
ans =

  logical

   0

Comparing H and H_23=r_x_alpha*trform_x_a*r_z_theta*trform_z_d
ans =

  logical

   0

Comparing H and H_24=r_x_alpha*trform_x_a*trform_z_d*r_z_theta
ans =

  logical

   0
```

Published with MATLAB® R2018a