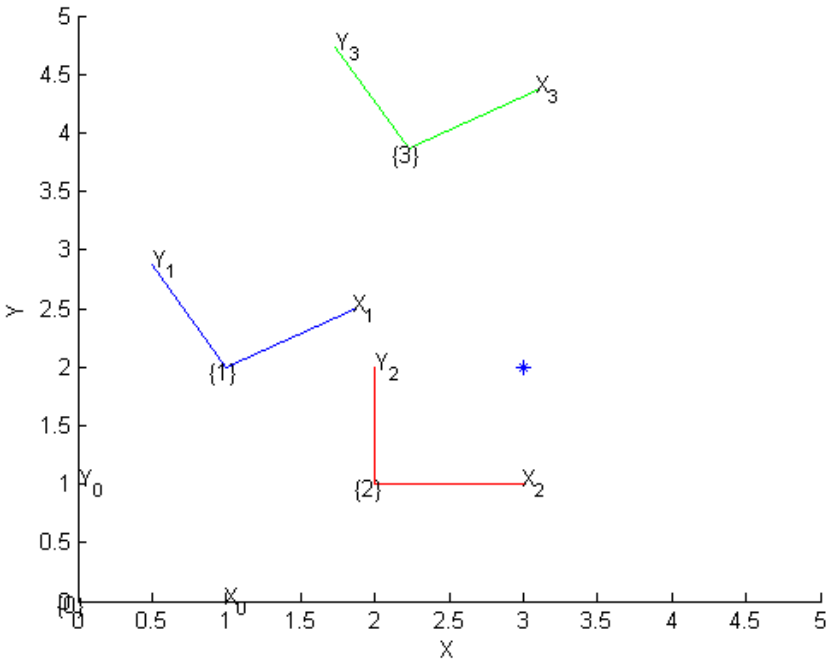
Robotics Toolbox, Position and Orientations

[1] Homogeneous Transformation in 2D:

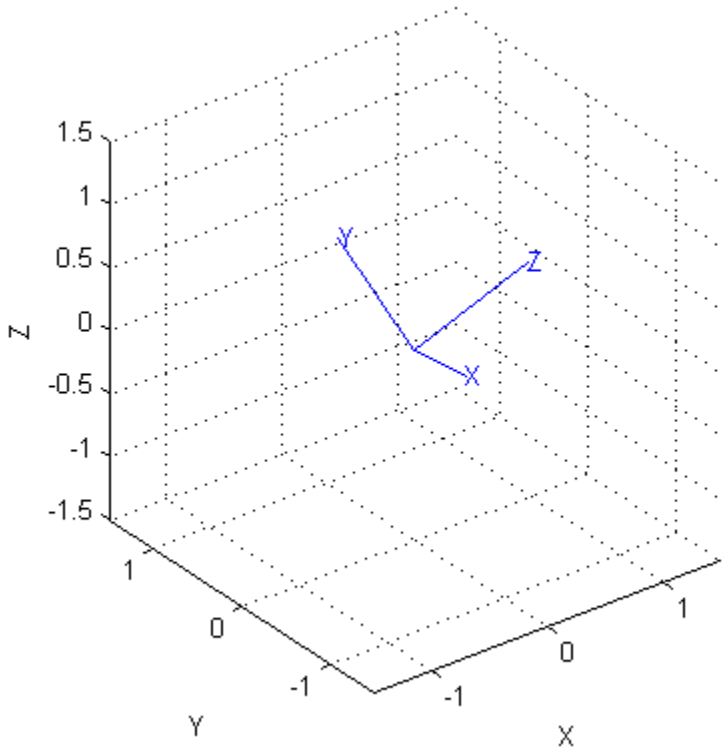
```
T1 = se2(1, 2, 30*pi/180); % which represents a translation of (1, 2)
and a rotation of 30°.
% To plot the T1 w.r.t world frame {0}:
figure(1);
trplot2(eye(3),'frame','0','color','c') % frame {0} worldframe
hold on;
trplot2(T1,'frame','1','color','b') % Blue frame {1}
axis([0 5 0 5]); % [x_min x_max y_min y_max] axis length.

% To multipy succesive transformations:
T2 = se2(2, 1, 0);
T3 = T1*T2; % transformation 1 then transformation  2
trplot2(T2,'frame','2','color','r') % Red frame {2}
trplot2(T3,'frame','3','color','g') % Green frame {3}
P = [3; 2]; % A point in 2D w.r.t frame {0}
plot_point(P, '*');
% To determine the coordinate of the point with respect to {1}
P1 = inv(T1) *[P; 1]; % Point P w.r.t frame {1}
```



[2] Basic Rotations:

Rotation about x by angle theta=90°:

```
Rx  = rotx(pi/2); % or R  = rotx(90,'deg');
% Rotation about y by angle theta=-60 deg:
Ry  = roty(-pi/3); % or R  = roty(-60,'deg');
% Rotation about z by angle theta=45 deg:
Rz  = rotz(pi/4); % or R  = rotz(45,'deg');
% To do successive basic rotations:
R = rotx(30,'deg')*roty(45,'deg');
% To plot the rotated coordinate:
figure(2);
trplot(R);
% To animate the rotation:
tranimate(R);
```



[3] Euler Angles Representation:

```
R1 = eul2r(0.1, 0.2, 0.3); % convert euler (phi, theta, psi) in rad to R
matrix.
% To convert R matrix to euler angles:
euler_angles = tr2eul(R1);
```

[4] RPY Representation:

```
R2 = rpy2r(0.1, 0.2, 0.3); % convert rpy (phi, theta, psi) in rad to R
matrix.
% To convert R matrix to RPY angles:
rpy_angles = tr2rpy(R2);
```

Homogeneous Transformation in 3D:

```
T1 = transl(1, 2, 3); %translation by a b c values
Rotation about x in 4x4:
T2 = trotx(pi/2); % in radians.
% Composite transformations:
T = transl(4,4,.5)*trotz(180,'deg')*troty(-30,'deg')*transl(0,0,.8);
figure(3);
trplot(eye(4),'frame','{0}','color','b','length',0.8); % original frame
hold on;
axis([0 5 0 5 0 2]);
tranimate(T,'color','r', 'frame', '{1}'); % transformed frame
```