

# 22AIE214-INTRODUCTION TO ROBOTICS

## LABSHEET 1

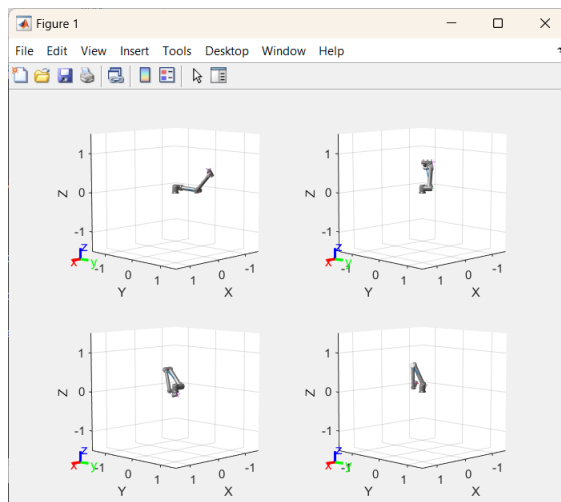
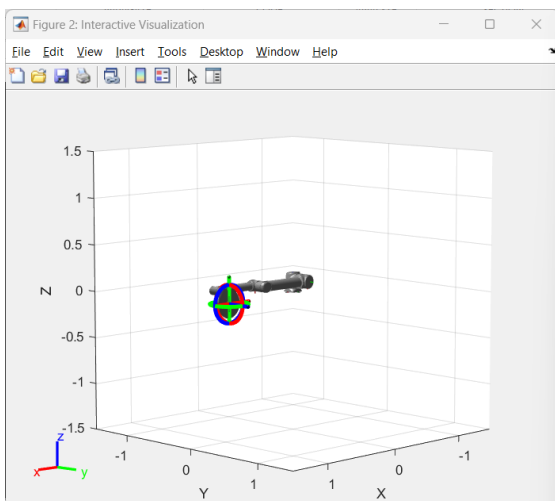
Name : Anuvind M P

Roll no: AM.EN.U4AIE22010

1. Using Matlab, load commercially available robot models like the Universal Robots™ UR10 robot, Boston Dynamics™ Atlas humanoid, and KINOVA™ Gen 3 manipulator. Explore how to generate joint configurations and interact with the robot models. [three separate programs , output screen shots required]

- universalUR10

```
ur10 = loadrobot("universalUR10");
show(ur10);
%Generating joint configurations
for i=1:4
    subplot(2,2,i);
    config = randomConfiguration(ur10);
    show(ur10,config)
end
% interACTING WITH ROBOT MODEL
interactiveGUI = interactiveRigidBodyTree(ur10);
% Saving the joint Configurations
interactiveGUIConfiguration = randomConfiguration(ur10);
addConfiguration(interactiveGUI);
disp(interactiveGUI.storedConfiguration);
```



- kinovaGen3

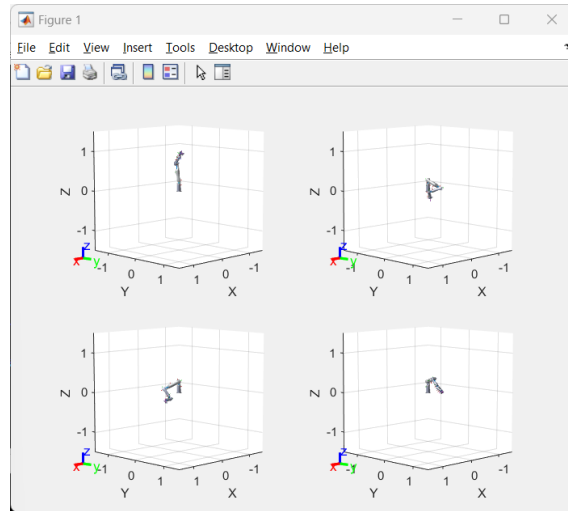
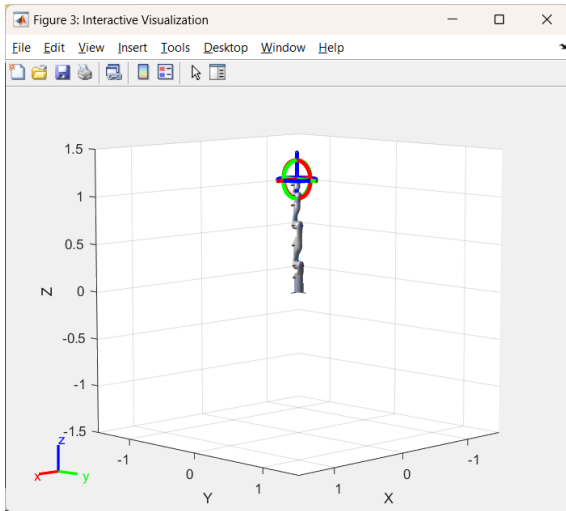
```
gen3 = loadrobot("kinovaGen3","DataFormat","column");
disp(gen3);
show(gen3);

%Generating joint configurations
for i=1:4
```

```

subplot(2,2,i);
config = randomConfiguration(gen3);
show(gen3,config)
end
% interACTING WITH ROBOT MODEL
interactiveGUI = interactiveRigidBodyTree(gen3);
% Saving the joint Configurations
interactiveGUIConfiguration = randomConfiguration(gen3);
addConfiguration(interactiveGUI);
disp(interactiveGUI.storedConfiguration);

```



## - Atlas :**

```

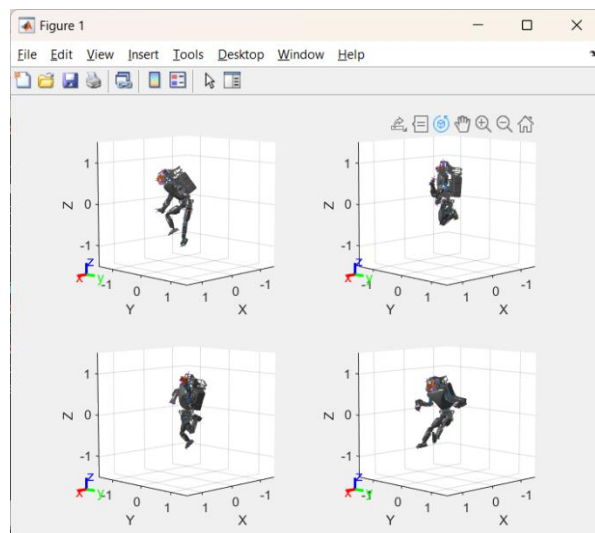
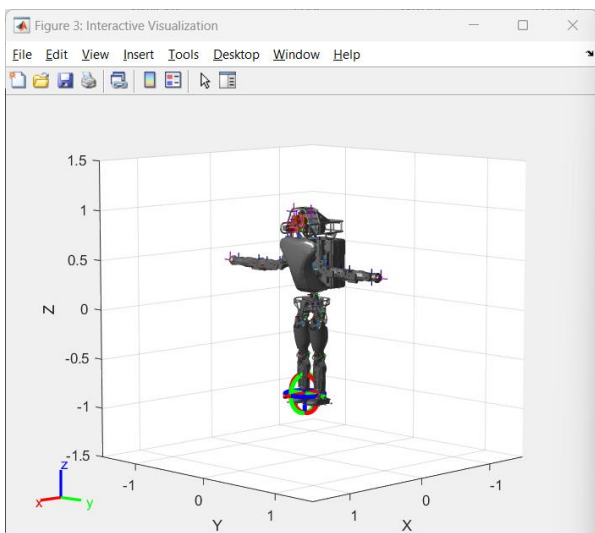
atlas = loadrobot("atlas");
show(atlas);

```

```

% Generating joint configurations
for i=1:4
    subplot(2,2,i);
    config = randomConfiguration(atlas);
    show(atlas,config)
end
% interACTING WITH ROBOT MODEL
interactiveGUI = interactiveRigidBodyTree(atlas);
% Saving the joint Configurations
interactiveGUIConfiguration = randomConfiguration(atlas);
addConfiguration(interactiveGUI);
disp(interactiveGUI.storedConfiguration);

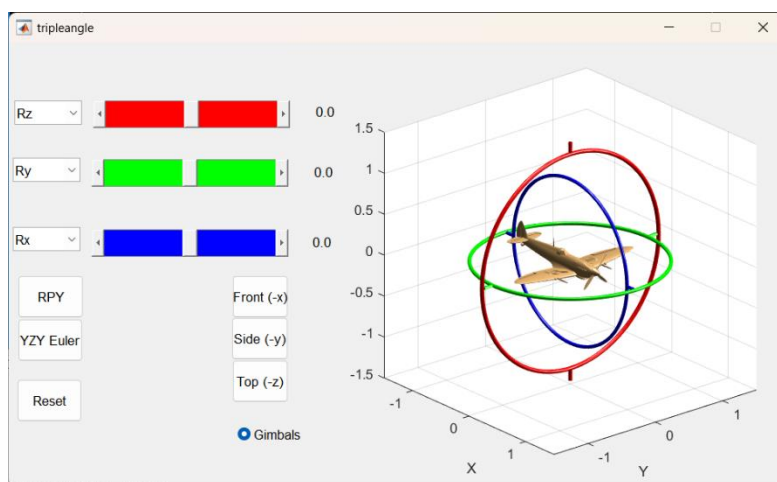
```



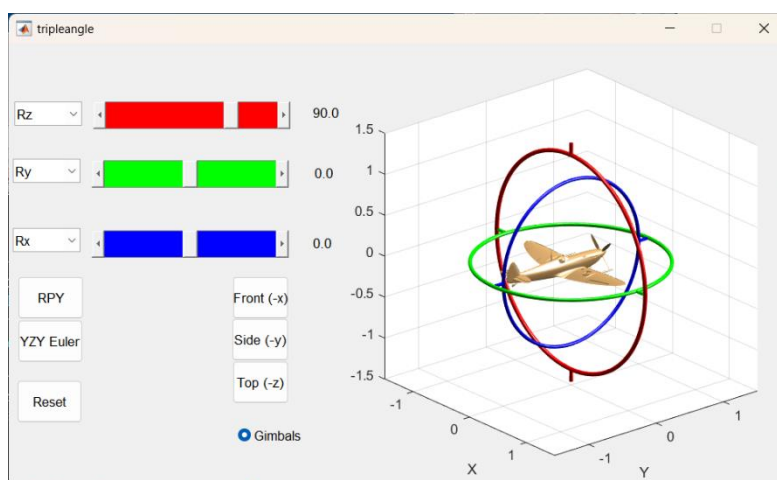
2. What is Gimbal lock . Explain in 10 sentences. Experiment with the triple angle in Matlab robotics toolbox. Explore roll pitch and yaw motions. [Command, screenshot of default model and the screenshot of gimbal lock model.]

### Gimbal lock :

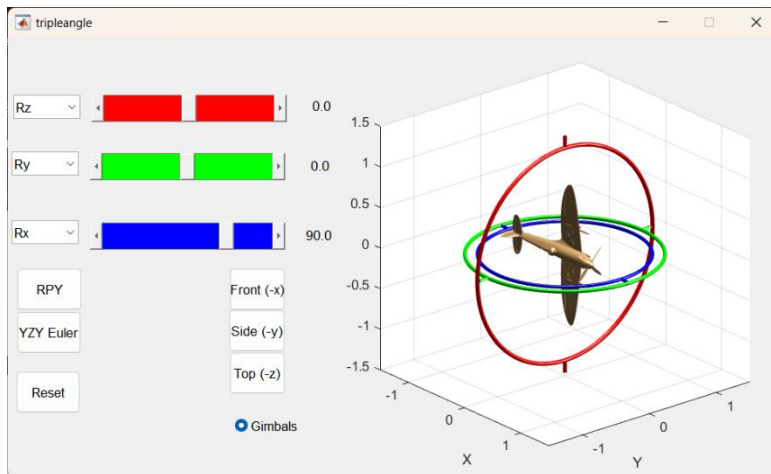
Gimbal lock is a phenomenon that occurs in a three-dimensional rotation system, like those used in aerospace or 3D graphics. Imagine a gyroscope with three gimbals, each representing a rotation around a different axis: pitch, yaw, and roll. When two of these axes align, it creates a loss of one degree of freedom, resulting in gimbal lock. This means that certain orientations become unreachable, limiting the range of motion. Gimbal lock commonly occurs when the system's pitch angle approaches  $\pm 90$  degrees, causing the gimbal representing yaw to align with the one representing roll. This can lead to unpredictable behavior in systems relying on precise orientation control. To mitigate gimbal lock, alternative rotation representations like quaternions are often used, as they don't suffer from this issue. Understanding gimbal lock is crucial in fields like aerospace, robotics, and computer graphics to ensure accurate and reliable motion control.



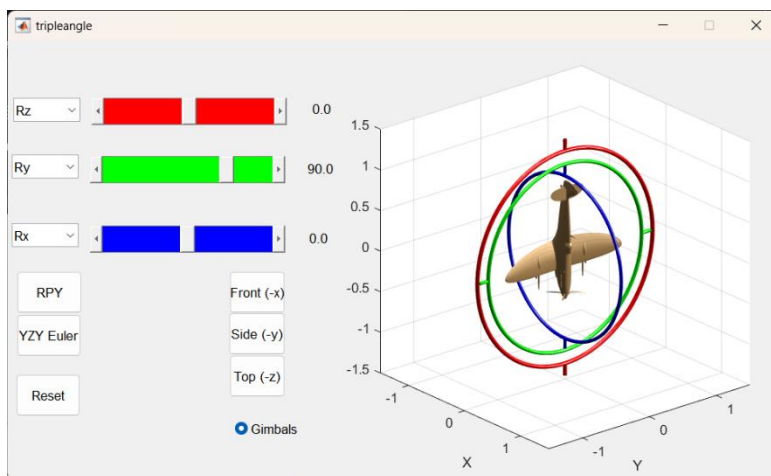
Default



**Yaw** : Yaw describes the rotation of an object around its vertical axis, typically passing through its center. It's like turning your head from side to side. In aviation, yawing involves a side-to-side movement of the aircraft's nose, with yawing to the left or right.



**Roll** : Roll involves the rotation of an object around its longitudinal axis, which runs from the front to the back. It's akin to tilting your head from one shoulder to the other. In aviation, rolling refers to the tilting motion of an aircraft's wings, with rolling to the left or right.



**Pitch** : Pitch refers to the rotation of an object around its lateral axis, which runs from side to side. Imagine nodding your head up and down; this motion is similar to pitch. In aviation, pitching up means the nose of the aircraft moves up relative to the horizon, while pitching down means the nose moves down.