Lab Number 09
MATLAB – Rigid Body Tree

Introduction:

CoppeliaSim EDU provides remote API to interface with different platforms such as ROS, MATLAB etc.

Software Used:

MATLAB R2021b

Programming Language:

MATLAB

Objectives of the Lab:

- To study rigid body tree and its uses.

RIGID BODY TREE:

A rigid body tree model is made up of rigid bodies as rigidBody objects. Each rigid body has a rigidBodyJoint object associated with it that defines how it can move relative to its parent body. Use setFixedTransform to define the fixed transformation between the frame of a joint and the frame of one of the adjacent bodies. You can add, replace, or remove rigid bodies from the model using the methods of the RigidBodyTree class.

Use the Denavit-Hartenberg (DH) parameters of the Puma560 robot to build a robot. Each rigid body is added one at a time, with the child-to-parent transform specified by the joint object.

The DH parameters define the geometry of the robot with relation to how each rigid body is attached to its parent. For convenience, setup the parameters for the Puma560 robot in a matrix. The Puma robot is a serial chain manipulator. The DH parameters are relative to the previous row in the matrix, corresponding to the previous joint attachment.

Create a rigid body tree object to build the robot.

```
robot = rigidBodyTree;
```

Create the first rigid body and add it to the robot. To add a rigid body:

1. Create a rigidBody object and give it a unique name.
2. Create a rigidBodyJoint object and give it a unique name.
3. Use setFixedTransform to specify the body-to-body transformation using DH parameters. The last element of the DH parameters, theta, is ignored because the angle is dependent on the joint position.
4. Call addBody to attach the first body joint to the base frame of the robot.

```
body1 = rigidBody('body1');
jnt1 = rigidBodyJoint('jnt1','revolute');

setFixedTransform(jnt1,dhparams(1,:),'dh');
body1.Joint = jnt1;

addBody(robot,body1,'base')
```

Create and add other rigid bodies to the robot. Specify the previous body name when calling addBody to attach it. Each fixed transform is relative to the previous joint coordinate frame.

```
body2 = rigidBody('body2');
jnt2 = rigidBodyJoint('jnt2','revolute');
body3 = rigidBody('body3');
jnt3 = rigidBodyJoint('jnt3','revolute');
body4 = rigidBody('body4');
jnt4 = rigidBodyJoint('jnt4','revolute');
body5 = rigidBody('body5');
jnt5 = rigidBodyJoint('jnt5','revolute');
body6 = rigidBody('body6');
jnt6 = rigidBodyJoint('jnt6','revolute');
```

```
setFixedTransform(jnt2,dhparams(2,:),'dh');
setFixedTransform(jnt3,dhparams(3,:),'dh');
setFixedTransform(jnt4,dhparams(4,:),'dh');
setFixedTransform(jnt5,dhparams(5,:),'dh');
setFixedTransform(jnt6,dhparams(6,:),'dh');

body2.Joint = jnt2;
body3.Joint = jnt3;
body4.Joint = jnt4;
body5.Joint = jnt5;
body6.Joint = jnt6;

addBody(robot,body2,'body1')
addBody(robot,body3,'body2')
addBody(robot,body4,'body3')
addBody(robot,body5,'body4')
addBody(robot,body6,'body5')
```

Now show the robot

```
show(robot);
axis([-0.5,0.5,-0.5,0.5,-0.5,0.5])
```

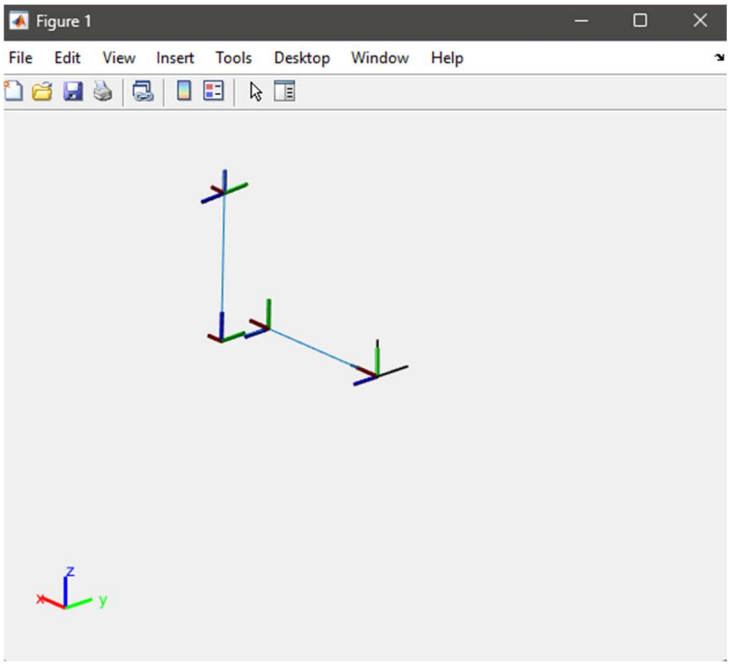
Figure 1
File Edit View Insert Tools Desktop Window Help
Z
x
y