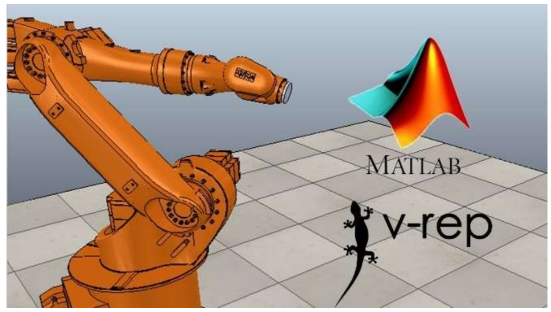
The First Glance on the Motion of a Robot

With *CoppeliaSim* Simulator



About this CoppeliaSim

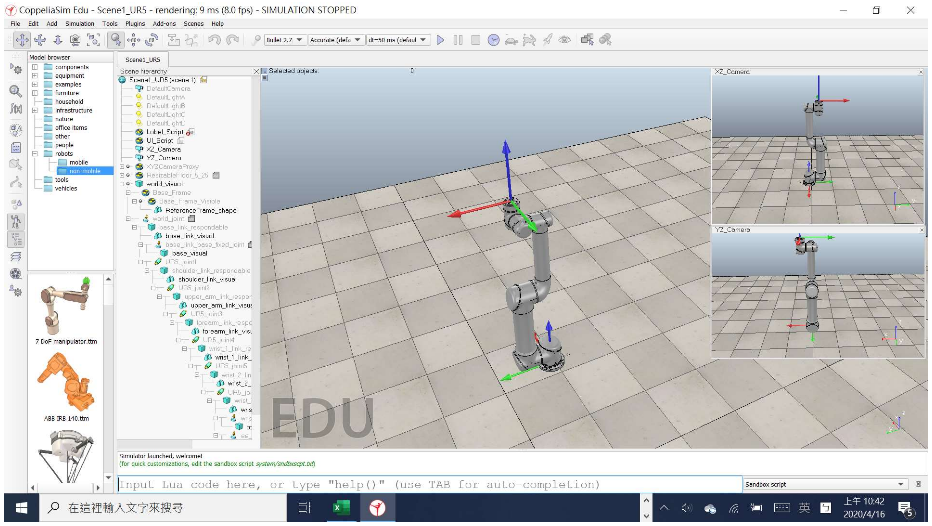
Without mathematical expression and programming understanding, it is never easy to tell and explain in great details how a robot moves in a certain direction and to a determined position. This simulating distribution, CoppeliaSim which is also called V-REP (the Virtual Robot Experimentation Platform of a 3D robot simulation software), clearly demonstrates the specific movement of a robot with intricacies and details. That is very helpful to make up our own minds of how to implement a robot with our desired way to use it properly and logically. Therefore, I am thrilled to apply this software in order to realize the complexity and wonder of how a robot works out in the way that is supposed to work out for the achievement of a smooth movement and goal.

In one example, I just want to briefly share my discover and my thought about the first time using this unique software on a robot. And I find that, through visual inspection, the forward kinematics can be calculated and interpreted and confirmed accordingly. I would like to post the screenshots of how this software can do and offer for the purpose of the motion of a robot in a mathematically and graphically and physically explanatory way. Since this is my first time examining the motion with this software, this is a good thing to get familiar with it through the small exercise on the Universal Robots UR5 robot, a popular 6-joint robot. At the desired configuration, the forward kinematics problem can be dealt with through CoppeliaSim. A formula called the "product of exponentials", which is the use of the matrix exponentials, can solve this problem as well. Hopefully, the experimentation of a robotic tryout is clearly displayed here.

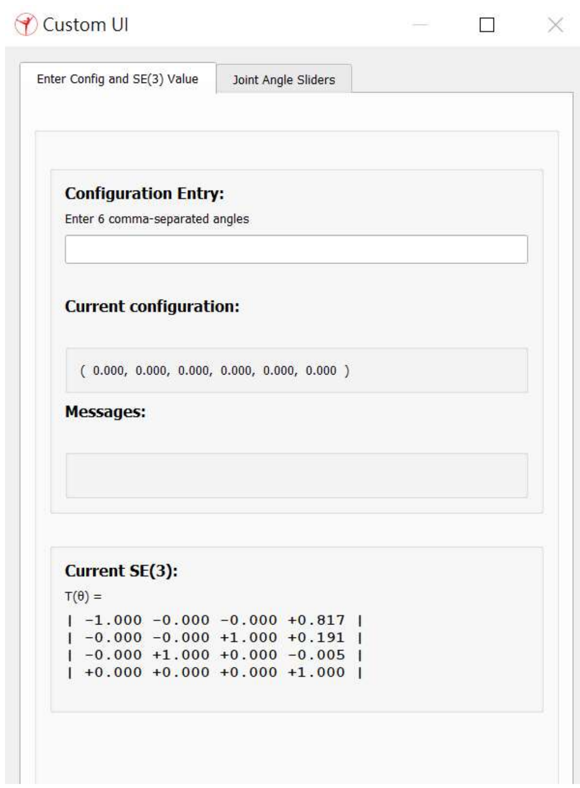
The goal is to find the configuration of the robot's end-effector in $T(\theta)\in SE(3)$ given the vector of positions of the joints, θ. Then the procedure is shown step-by-step below.

Procedure

This is the website I go to, and implement Scene 1 as my first demo example, which is the UR5 robot as shown below.

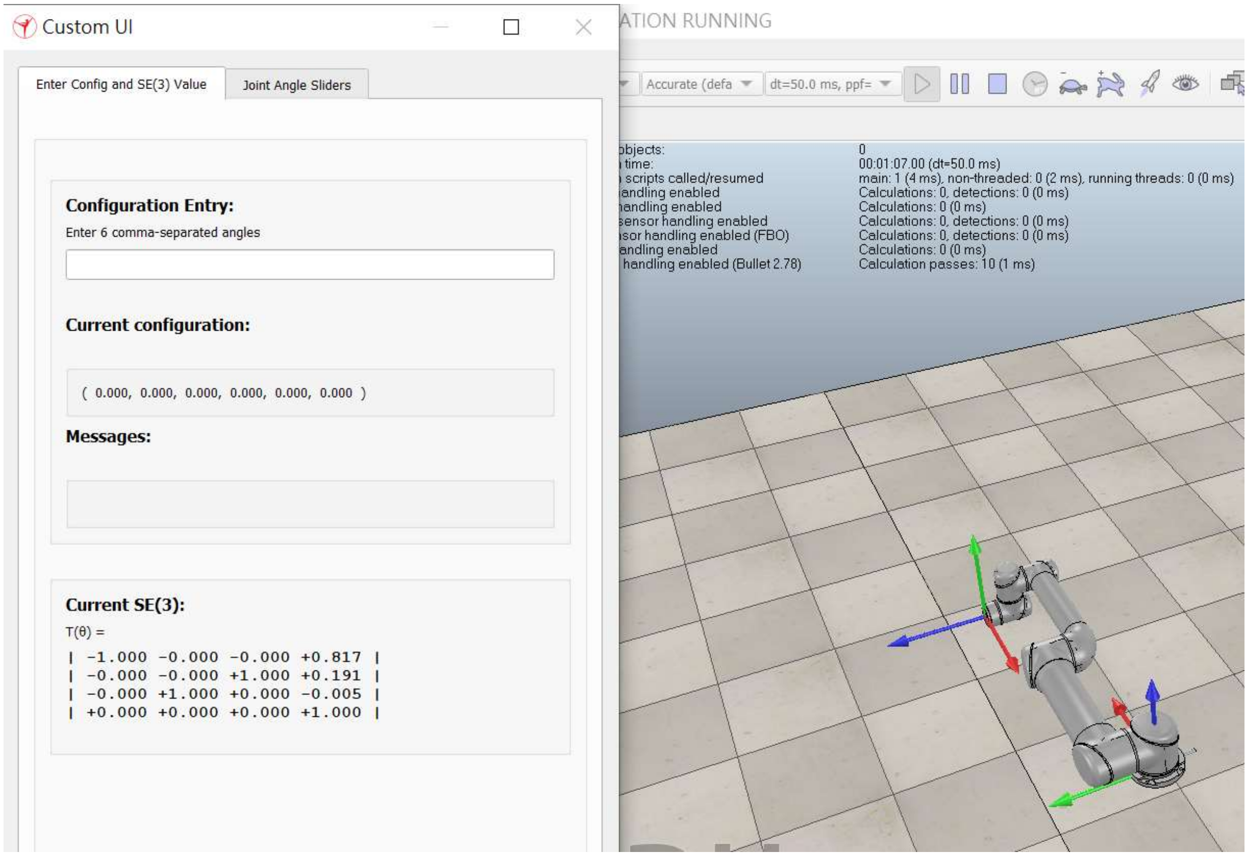


If we run the scene, this window with two tabs, Enter Config and SE(3) Value and Joint Angle Sliders, will come up:
(SE: Special Euclidean)

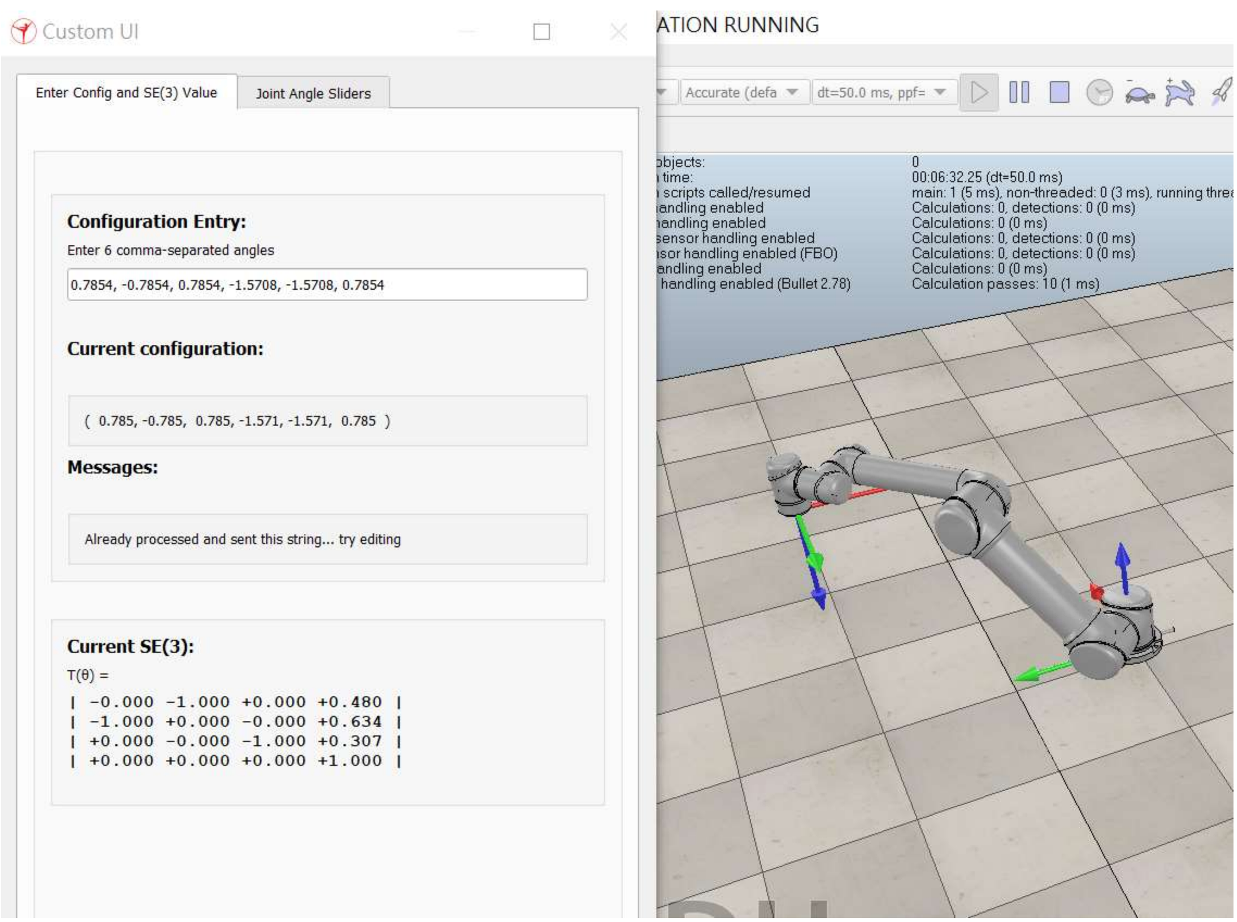


Since this is the forward kinematics problem, the joint angles are known. In that case, I want to set up the values for joint angles to 0.7854, -0.7854, 0.7854, -1.5708, -1.5708, 0.7854. (This is a 6-joint robot, so six values) Then type these values into the box of "Configuration Entry". After the values are entered and the program runs, the end-effector frame configuration relative to the frame at the base of the robot is seen, and this will be reported as the matrix $T(\theta)$. By the way, for each frame, the x-axis is red, y-axis is green, and z-axis is blue. The result is like the image below:

The original configuration (0, 0, 0, 0, 0, 0):

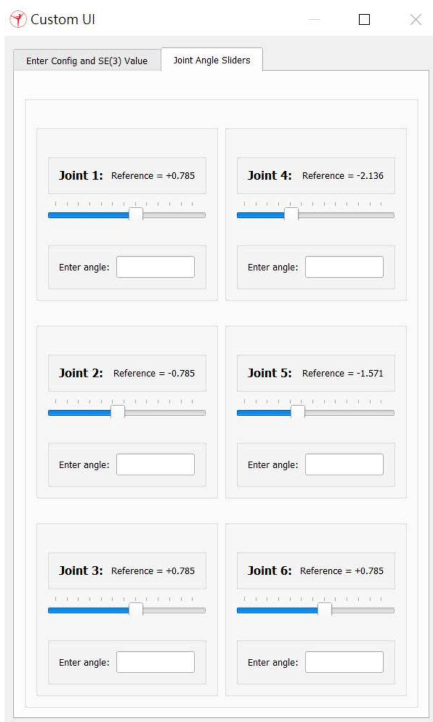


The Configuration Entry: (0.7854, -0.7854, 0.7854, -1.5708, -1.5708, 0.7854):



As clearly seen, both the {s} and {b} coordinates and the window displaying the transformation matrix SE(3) are shown. The matrix also points us to see the relationship between the 3 axes of the {s} frame and the 3 axes of the {b} frame and a vector values based on the {s} coordinates. Therefore, this is an easy way to look at the transformation movement on a robot.

For extra information, if we go to the tag of "Join Angle Sliders", we can control each joint angle as well. the beneficial thing about this use is to be able to help us watch the movement of each angle closely and clearly by implementing the sliding part.



In the end, there are tons of robots that are worthy for me to keep exploring for the understanding and uniqueness of more different kinds of ways to move them. This is just one of the examples I exercise myself as my initial step. But overall, this is a good software, CoppeliaSim that drives me into the deeper sense of grasping the knowledge of the motion of a robot. And of course, I quite enjoy it for the hands-on learning experience like this.