

Curso: Robótica IELE-3338
Semestre: 2022-10
Profesor: Juan José García
Asistentes: Susana Marcela Chavez
Monitores: Vilma Tirado
Entrega: No entregable



Actividad en clase Planeación de trayectorias



Figure 1: Manipulador Panda, by Emika Franka, más información en: <https://www.franka.de/>

Enunciado

1. **Motion Planning:** Para ejecutar la aplicación de planeación de ruta del manipulador debe ejecutar en una terminal el simulador del manipulador Panda:

```
roslaunch panda_moveit_config demo.launch
```

Y en una terminal nueva ejecutar el nodo de planeación de ruta de la siguiente forma:

```
roslaunch moveit_tutorials motion_planning_api_tutorial.launch
```

Vea el resultado y siga las instrucciones que se imprimen en esa terminal.

Estudie y modifique el código del archivo *motion_planning_api_tutorials.cpp* de forma que a partir de ciertos goals personalizados puedan ejecutar la planeación de ruta y se vea como va cambiando la

misma. El objetivo es que explique de forma detallada como funciona este nodo de ROS y lo ponga a prueba con posiciones finales deseadas (Mínimo dos goals de prueba). Siga el siguiente ejemplo:

Ejemplo: En el archivo *motion_planning_api_tutorial.cpp* se encuentran programadas en C++ las funcionalidades del nodo previamente descritas. A partir de la línea 150 se establece el goal al que se quiere llegar por medio de planeación de ruta, tal y como se muestra en las siguientes líneas de código:

```
159     pose.header.frame_id = "panda_link0";
160     pose.pose.position.x = 0.5;
161     pose.pose.position.y = 0.5;
162     pose.pose.position.z = 1.0;
163     pose.pose.orientation.w = 1.0;
```

Estas son algunas de las líneas que se debe cambiar para fijar un nuevo goal. Sin embargo, a lo largo de la aplicación se pueden evidenciar otros goals. Usted debe revisar e iterar sobre estos nuevos goals.

2. **Pick and Place scene:** Para ejecutar la aplicación de Pick and Place del manipulador debe ejecutar en una terminal el simulador del manipulador Panda:

```
roslaunch panda_moveit_config demo.launch
```

Y en una terminal nueva ejecutar el nodo de la aplicación en cuestión:

```
roslaunch moveit_tutorials pick_place_tutorial
```

Vea el resultado y siga las instrucciones que se imprimen en esa terminal.

Estudie y modifique el código del archivo *pick_and_place_tutorial.cpp* de forma que a partir de las dimensiones y posiciones de las mesas y el objeto a manipular se puedan ver diversas opciones de esta aplicación. El objetivo es que explique de forma detallada como funciona este nodo de ROS y lo ponga a prueba variando los parámetros de los mismos. Siga el siguiente ejemplo:

Ejemplo: En el archivo *pick_and_place_tutorial.cpp* se encuentran programadas en C++ las funcionalidades del nodo previamente descritas. En este archivo se encuentra divididos por métodos cada una de las acciones que debe hacer el manipulador: desde su posición inicial y posición final, hasta las dimensiones y posiciones de los objetos auxiliares. En el método de `addCollisionObjects` se encuentran todas los objetos que se han adicionado a la escena (Objeto a manipular y las mesas)

3. En la última aplicación de Pick And Place modifique el código para que el robot lleve el objeto de la mesa final a una mesa adicional. Es decir, adicione una mesa a la escena y haga que el robot lo lleve hasta esta nueva mesa final. De esta manera al finalizar se tendrá 3 planeaciones de ruta diferentes.