

Le Robot Marcheur

PROJECT ROBOTIQUE INDUSTRIELLE

SUNGKUR SEEKCHA

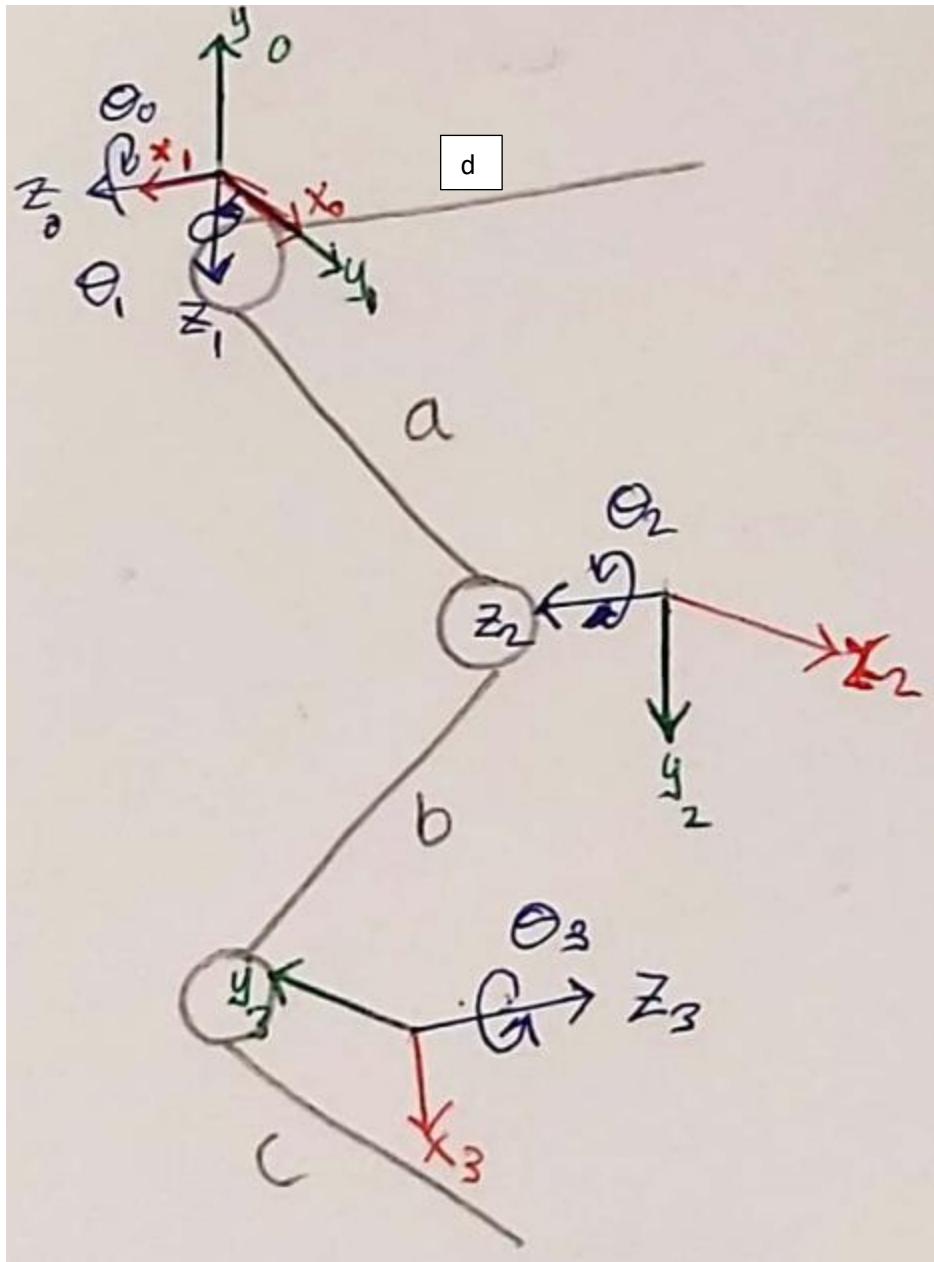
Table of Contents

| | |
|---------------------------------------|----|
| Table of Contents..... | i |
| Partie 1 Conception d'une jambe | 1 |
| Direct movement : | 10 |
| Indirect movement : | 11 |
| Annex | l |

Partie 1 Conception d'une jambe

1. Concevoir le schéma cinématique de la jambe de votre système.





2. Indiquer les repères liés aux différents solides de votre système.

See in the picture above straight-line a, b, c and d. The circles are the revolute joint.

3. Déterminer la matrice de Denavit-Hartenberg.

| n | θ | α | r | d |
|---|------------|----------|---|---|
| 1 | 180 | 90 | 0 | 0 |
| 2 | θ_2 | 0 | a | 0 |
| 3 | 90 | 180 | 0 | 0 |

4. Créer un package qui contiendra l'urdf de votre robot ainsi que les launchfile nécessaires à la visualisation des urdf.

```
deeya@deeya-VirtualBox:~/catkin_ws$ cd src
deeya@deeya-VirtualBox:~/catkin_ws/src$ catkin_create_pkg urdf_quad roscpp
std_msgs
Created file urdf_quad/package.xml
Created file urdf_quad/CMakeLists.txt
Created folder urdf_quad/include/urdf_quad
Created folder urdf_quad/src
Successfully created files in /home/deeya/catkin_ws/src/urdf_quad. Please adjust
the values in package.xml.
deeya@deeya-VirtualBox:~/catkin_ws/src$ ls
CMakeLists.txt    udm_hand_moveit_configs  udm_urdf
udm_hand_control  udm_leg                  urdf_quad
deeya@deeya-VirtualBox:~/catkin_ws/src$ cd urdf_quad
deeya@deeya-VirtualBox:~/catkin_ws/src/urdf_quad$ ls
CMakeLists.txt  include  package.xml  src
deeya@deeya-VirtualBox:~/catkin_ws/src/urdf_quad$ mkdir launch
deeya@deeya-VirtualBox:~/catkin_ws/src/urdf_quad$ cd launch
deeya@deeya-VirtualBox:~/catkin_ws/src/urdf_quad/launch$ touch check_urdf.launch
deeya@deeya-VirtualBox:~/catkin_ws/src/urdf_quad/launch$ ls
check_urdf.launch
deeya@deeya-VirtualBox:~/catkin_ws/src/urdf_quad/launch$
```

```
check_urdf.launch
~/catkin_ws/src/urdf_quad/launch

<launch>

  <arg name="model" default="$(find urdf_quad)/urdf/main.urdf"/>
  <param name="robot_description" command="$(find xacro)/xacro.py $(arg model)" />

  <!-- Combine joint values -->
  <node name="joint_state_publisher_gui" pkg="joint_state_publisher_gui"
type="joint_state_publisher_gui" />

  <!-- Send joint values -->
  <node name="robot_state_publisher" pkg="robot_state_publisher" type="robot_state_publisher" />

  <!-- Show in Rviz -->
  <node name="rviz" pkg="rviz" type="rviz" args="-d $(find urdf_quad)/rviz/urdf.rviz"/>

</launch>
```

5. Créer l'urdf de votre robot avec une jambe et le base_link de votre robot.

```
deeya@deeya-VirtualBox:~/catkin_ws$ source devel/setup.bash
deeya@deeya-VirtualBox:~/catkin_ws$ roslaunch urdf_quad check_urdf.launch
... logging to /home/deeya/.ros/log/cba8dd10-047a-11eb-9877-080027eb1150/roslaunch-deeya-VirtualBox-2452.log
Checking log directory for disk usage. This may take a while.
Press Ctrl-C to interrupt
Done checking log file disk usage. Usage is <1GB.

xacro.py is deprecated; please use xacro instead
started roslaunch server http://deeya-VirtualBox:35887/

SUMMARY
=====

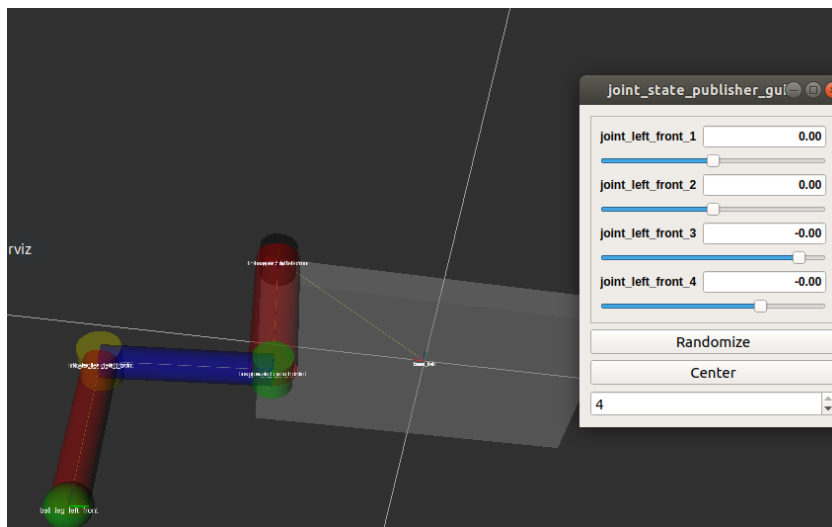
PARAMETERS
* /robot_description: <?xml version="1....
* /rostdistro: melodic
* /rosversion: 1.14.9

deeya@deeya-VirtualBox:~/catkin_ws$ cd src/urdf_quad
deeya@deeya-VirtualBox:~/catkin_ws/src/urdf_quad$ ls
CMakeLists.txt  include  launch  package.xml  src
deeya@deeya-VirtualBox:~/catkin_ws/src/urdf_quad$ mkdir urdf
deeya@deeya-VirtualBox:~/catkin_ws/src/urdf_quad$ ls
CMakeLists.txt  include  launch  package.xml  src  urdf
deeya@deeya-VirtualBox:~/catkin_ws/src/urdf_quad$ cd urdf
deeya@deeya-VirtualBox:~/catkin_ws/src/urdf_quad/urdf$ touch main.urdf
deeya@deeya-VirtualBox:~/catkin_ws/src/urdf_quad/urdf$ ls
main.urdf
```

```

deeya@deeya-VirtualBox:~/catkin_ws$ catkin build
-----
Profile:                                default
Extending:                             [cached] /opt/ros/melodic
Workspace:                              /home/deeya/catkin_ws
-----
Build Space:                            [exists] /home/deeya/catkin_ws/build
Devel Space:                            [exists] /home/deeya/catkin_ws/devel
Install Space:                          [unused] /home/deeya/catkin_ws/install
Log Space:                              [exists] /home/deeya/catkin_ws/logs
Source Space:                           [exists] /home/deeya/catkin_ws/src
DESTDIR:                                [unused] None
-----
Devel Space Layout:                      linked
Install Space Layout:                   None
-----
Additional CMake Args:                  None
Additional Make Args:                   None
Additional catkin Make Args:            None
Internal Make Job Server:               True
Cache Job Environments:                 False
-----
Whitelisted Packages:                   None
Blacklisted Packages:                   None
-----
Workspace configuration appears valid.
-----
[build] Found '5' packages in 0.0 seconds.
[build] Updating package table.
Starting   >>> udm_hand_control
Finished   <<< udm_hand_control           [ 0.5 seconds ]
Starting   >>> udm_leg
Finished   <<< udm_leg                     [ 0.4 seconds ]
Starting   >>> udm_urdf
Finished   <<< udm_urdf                     [ 0.3 seconds ]
Starting   >>> urdf_quad
Finished   <<< urdf_quad                     [ 5.0 seconds ]
Starting   >>> udm_hand_moveit_configs
Finished   <<< udm_hand_moveit_configs       [ 0.3 seconds ]
[build] Summary: All 5 packages succeeded!
[build] Ignored: None.
[build] Warnings: None.
[build] Abandoned: None.
[build] Failed: None.
[build] Runtime: 6.7 seconds total.
[build] Note: Workspace packages have changed, please re-source setup files to use them.

```



6. Créer le package `udm_project_moveitconfig` avec le `moveit` assistant `setup`.

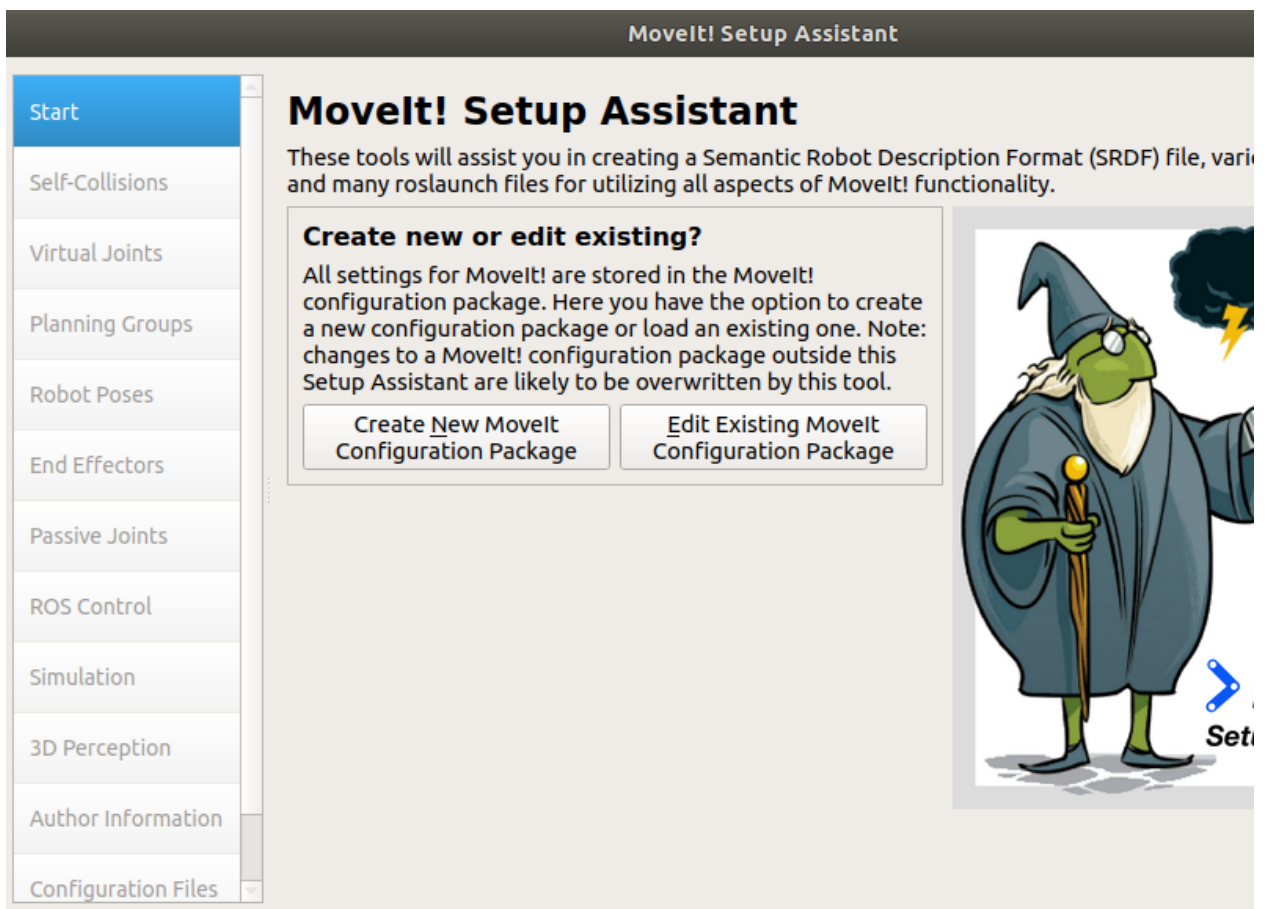
```
deeya@deeya-VirtualBox:~/catkin_ws$ roslaunch moveit_setup_assistant setup_assistant.launch
... logging to /home/deeya/.ros/log/f94acc24-0493-11eb-9877-080027eb1150/roslaunch-deeya-VirtualBox-5581.log
Checking log directory for disk usage. This may take a while.
Press Ctrl-C to interrupt
Done checking log file disk usage. Usage is <1GB.

started roslaunch server http://deeya-VirtualBox:32935/

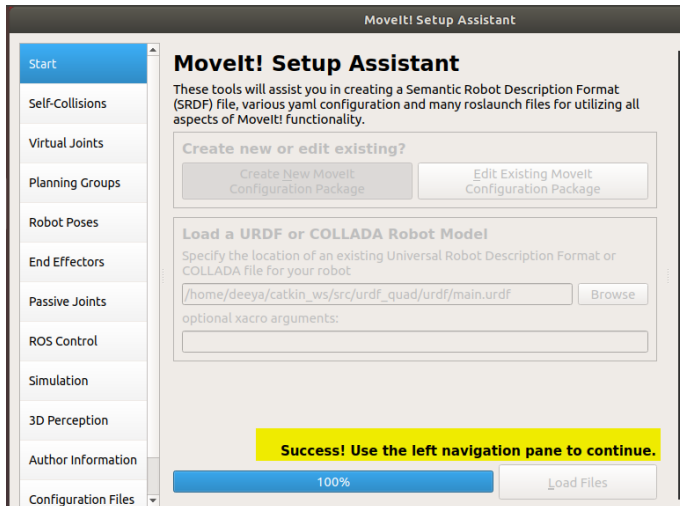
SUMMARY
=====
```

7. Créer un package `udm_project_control` avec un noeud permettant de contrôler la jambe de manière directe, puis un autre noeud contrôlant la jambe de manière indirecte.

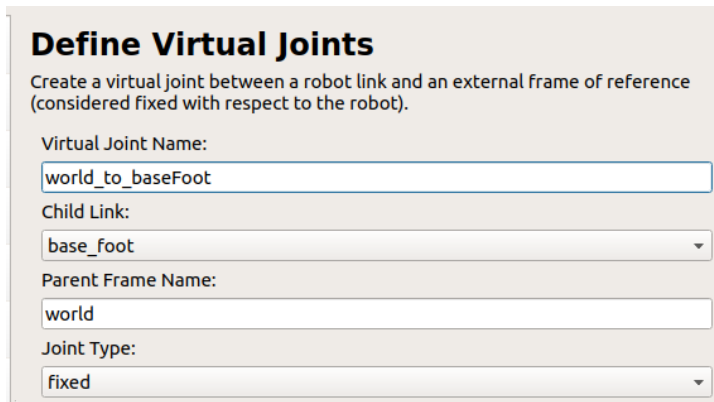
1. Click on create New and browse your urdf file



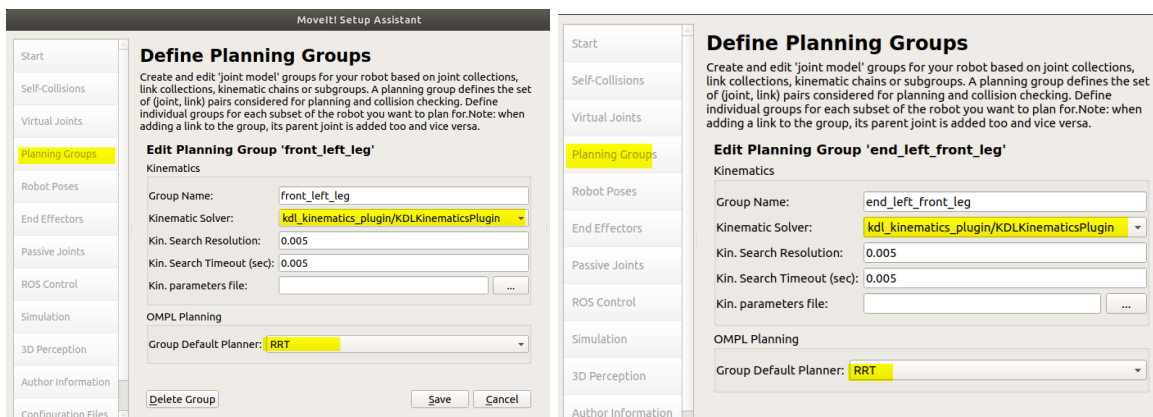
2. Click on load.



3. Add virtual joint



4. Click on planning groups.



- Click on add joints and select the joint.

Define Planning Groups

Create and edit 'joint model' groups for your robot based on joint collections, link collections, kinematic chains or subgroups. A planning group defines the set of (joint, link) pairs considered for planning and collision checking. Define individual groups for each subset of the robot you want to plan for. Note: when adding a link to the group, its parent joint is added too and vice versa.

Edit 'front_left_leg' Joint Collection

Available Joints

| Joint Names |
|------------------------------|
| 1 world_to_baseFoot |
| 2 base_link_joint |
| 3 joint_left_front_1 |
| 4 joint_left_front_2 |
| 5 upper_leg_left_front_joint |
| 6 joint_left_front_3 |
| 7 lower_leg_left_front_joint |
| 8 joint_left_front_4 |
| 9 ball_leg_left_front_joint |

Selected Joints

| Joint Names |
|------------------------------|
| 1 world_to_baseFoot |
| 2 base_link_joint |
| 3 joint_left_front_1 |
| 4 joint_left_front_2 |
| 5 upper_leg_left_front_joint |
| 6 joint_left_front_3 |
| 7 lower_leg_left_front_joint |
| 8 joint_left_front_4 |
| 9 ball_leg_left_front_joint |

Save

Cancel

Current Groups

- front_left_leg
 - Joints
 - world_to_baseFoot - Fixed
 - base_link_joint - Fixed
 - joint_left_front_1 - Revolute
 - joint_left_front_2 - Revolute
 - upper_leg_left_front_joint - Fixed
 - joint_left_front_3 - Revolute
 - lower_leg_left_front_joint - Fixed
 - joint_left_front_4 - Revolute
 - ball_leg_left_front_joint - Fixed
 - Links
 - Chain
 - base_foot -> ball_leg_left_front
 - Subgroups
- end_left_front_leg
 - Joints
 - ball_leg_left_front_joint - Fixed
 - Links
 - link_foot_leg_left_front
 - ball_leg_left_front
 - Chain
 - Subgroups

- Add robot pose.

Define Robot Poses

Create poses for the robot. Poses are defined as sets of joint values for particular planning groups. This is useful for things like *home position*. The first pose for each robot will be its initial pose in simulation.

Pose Name:

Planning Group:

Setup Assistant

joint_left_front_1

1.2100

joint_left_front_2

0.1963

joint_left_front_3

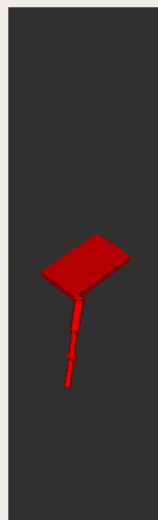
-1.6400

joint_left_front_4

-1.6299

Save

Cancel



- Add end effector.

Define End Effectors

Setup your robot's end effectors. These are planning groups corresponding to grippers or tools, attached to a parent planning group (an arm). The specified parent link is used as the reference frame for IK attempts.

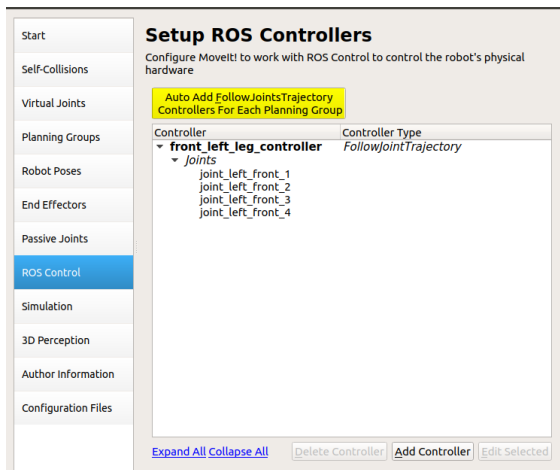
End Effector Name:

End Effector Group:

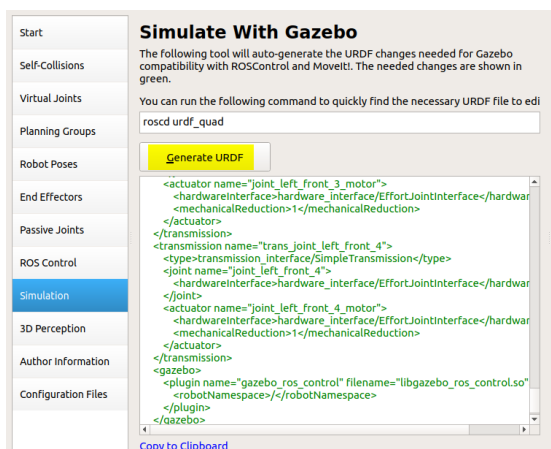
Parent Link (usually part of the arm):

Parent Group (optional):

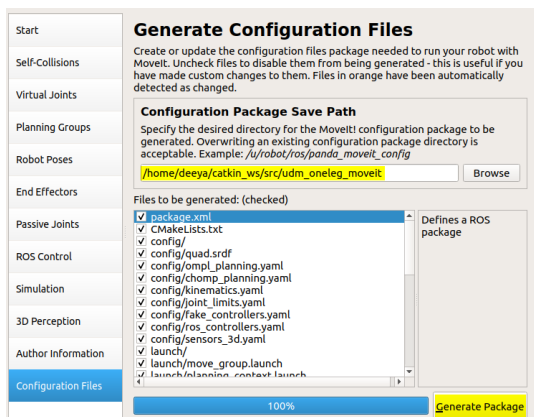
- Click on ROS Control > auto Add followJointsTrajectory Controllers for each planning group.



- Generate simulate with Gazebo.



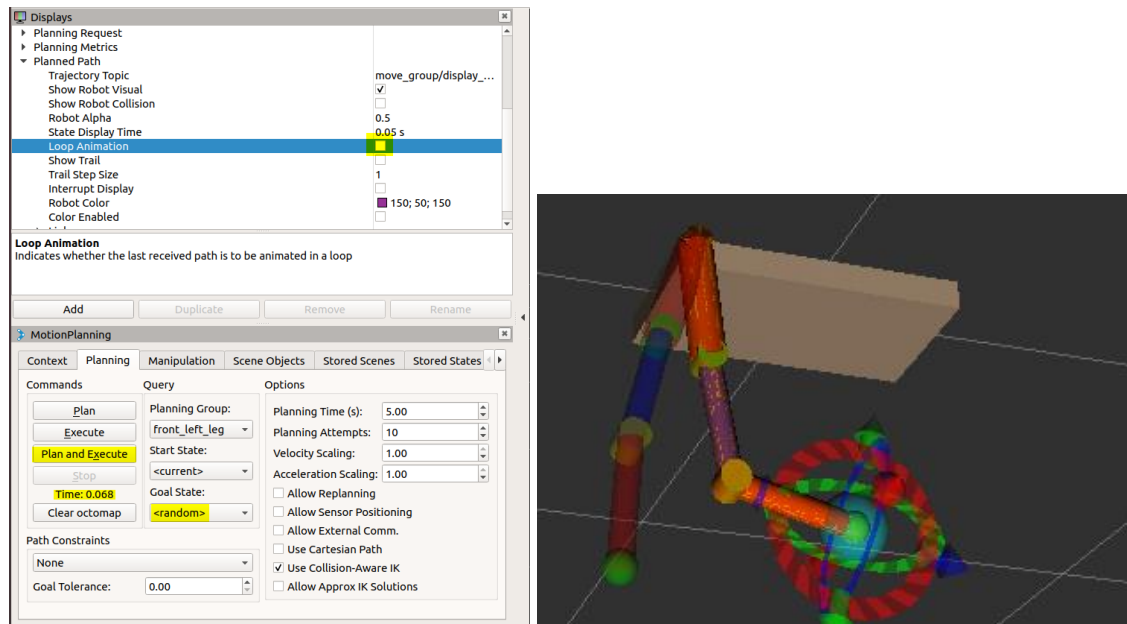
- Generate the package file.



11. Launch demo.launch

```
deeya@deeya-VirtualBox:~/catkin_ws$ roslaunch udm_oneleg_moveit demo.launch
... logging to /home/deeya/.ros/log/8cd838da-04e6-11eb-9877-080027eb1150/roslaun
ch-deeya-VirtualBox-19610.log
```

12. Plan and execute.

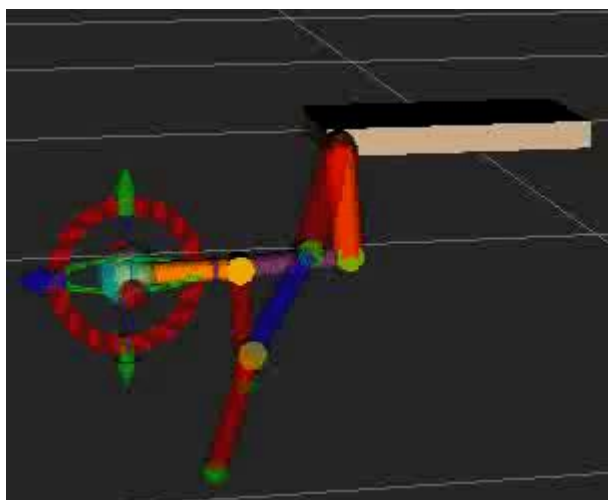


Direct movement :

1. Launch direct.launch

```
deeya@deeya-VirtualBox:~/catkin_ws$ roslaunch udm_project_control direct.launch
... logging to /home/deeya/.ros/log/f488b0b0-070a-11eb-b958-080027eb1150/roslaun
```

Click on image to see movement.



2. Parameters used to call function

```
deeya@deeya-VirtualBox:~/catkin_ws$ rosservice call /direct_kin_service_front_left_leg "joint1:
  data: 0.7100
joint2:
  data: 0.0
joint3:
  data: 0.0
joint4:
  data: 0.0"
res:
  data: True
message:
  data: "Success"
```

Indirect movement :

1. Launch indirect.launch

```
deeya@deeya-VirtualBox:~/catkin_ws$ roslaunch udm_project_control indirect.launch
... logging to /home/deeya/.ros/log/7c0a8204-05ab-11eb-915d-080027eb1150/roslaunch-deeya-Virtual
Box-20854.log
Checking log directory for disk usage. This may take a while.
Press Ctrl-C to interrupt
Done checking log file disk usage. Usage is <1GB.

started roslaunch server http://deeya-VirtualBox:33451/

SUMMARY
=====

PARAMETERS
* /move_group_leg/group_name: front_left_leg
* /rostdistro: melodic
* /rosversion: 1.14.9
```

2. Parameters used to call function

```
deeya@deeya-VirtualBox:~/catkin_ws$ rosservice call /indirect_kin_service_front_left_leg "pose:
orientation:
  w: 1
position:
  x: 0.01
position:
  y: 0.2
position:
  z: -0.01"
res:
  data: True
message:
  data: "Success"
```

Annex

1. Error: forgot to add other dependencies.

Solution: delete package.xml and creating a new one.

```
CMakeLists.txt udm_hand_control udm_hand_moveit_configs udm_leg udm_urdf urdf_quad
deeya@deeya-VirtualBox:~/catkin_ws/src$ cd urdf_quad
deeya@deeya-VirtualBox:~/catkin_ws/src/urdf_quad$ ls
CMakeLists.txt include package.xml src
deeya@deeya-VirtualBox:~/catkin_ws/src/urdf_quad$ rm package.xml
deeya@deeya-VirtualBox:~/catkin_ws/src/urdf_quad$ ls
CMakeLists.txt include src
deeya@deeya-VirtualBox:~/catkin_ws/src/urdf_quad$ cd
deeya@deeya-VirtualBox:~$ cd catkin_ws/src
deeya@deeya-VirtualBox:~/catkin_ws/src$ catkin_create_pkg urdf_quad rospy roscpp urdf std_msgs geometry_msgs sensor_msgs
```

```
Created file urdf_quad/package.xml
Created file urdf_quad/CMakeLists.txt
Created folder urdf_quad/include/urdf_quad
Created folder urdf_quad/src
Successfully created files in /home/deeya/catkin_ws/src/urdf_quad/urdf_quad. Please adjust the values in package.xml.
```

2. Error: cannot launch node

```
ERROR: cannot launch node of type [udm_project_control/direct_kin_service.py]: Cannot locate node of type [direct_kin_service.py] in package [udm_project_control]. Make sure file exists in package path and permission is set to executable (chmod +x)
```

Solution:

```
deeya@deeya-VirtualBox:~/catkin_ws/src/udm_project_control/src$ chmod +x direct_kin_service.py
deeya@deeya-VirtualBox:~/catkin_ws/src/udm_project_control/src$ cd
deeya@deeya-VirtualBox:~$ cd catkin_ws
deeya@deeya-VirtualBox:~/catkin_ws$ roslaunch udm_project_control direct.launch
... logging to /home/deeya/.ros/log/2eeefb14-05a4-11eb-915d-080027eb1150/roslaunch-deeya-VirtualBox-6061.log
Checking log directory for disk usage. This may take a while.
Press Ctrl-C to interrupt
Done checking log file disk usage. Usage is <1GB.

started roslaunch server http://deeya-VirtualBox:40543/

SUMMARY
=====
```

3. Error: unable to communicate with master

```

deeya@deeya-VirtualBox:~/catkin_ws$ rosservice
call /direct_kin_service_front_left_leg "joint1
:
> data: 1.21
> joint:
> data: 0.109
> joint:
> data: 0.17
> joint:
> data: -1.625"
ERROR: Unable to communicate with master!

```

Solution: Start roscore in other shell.

```

deeya@deeya-VirtualBox:~/catkin_ws$ roscore
... logging to /home/deeya/.ros/log/7c0a8204-05
ab-11eb-915d-080027eb1150/roslaunch-deeya-Virtu
alBox-6736.log
Checking log directory for disk usage. This may
take a while.
Press Ctrl-C to interrupt
Done checking log file disk usage. Usage is <1G
B.

```

4. Error: Robot model parameter not found

```

/
move_group_leg (udm_project_control/direct_
kin_service.py)

ROS_MASTER_URI=http://localhost:11311

process[move_group_leg-1]: started with pid [81
64]
[ERROR] [1601756019.871804801]: Robot model par
ameter not found! Did you remap 'robot_descript
ion'?
Traceback (most recent call last):
  File "/home/deeya/catkin_ws/src/udm_project_c
ontrol/src/direct_kin_service.py", line 62, in
<module>
    groupService()

```

Solution: Add the following parameters