

Movelt Inverse Kinematics Design

Movelt Inverse Kinematics Design

1. Usage Environment
2. Drive Real Robot
 - 2.1. Start Real Robot
 - 2.2. Launch Movelt2
3. Inverse Kinematics Design
 - Launch Command
 - Precautions

1. Usage Environment

Board: Jetson Orin Nano/Nx

ROS2: Humble

2. Drive Real Robot

Driving the real robot is done by subscribing to Movelt2's `/joint_states` topic and converting the robotic arm's joint state information into control of the real robotic arm.

Note: Since the real robotic arm does not have obstacle avoidance functionality, some poses may collide with obstacles; therefore, try to make the planned robotic arm motions reasonable and avoid positions with obstacles

(It is recommended to use preset poses for driving the real robot demonstration)

2.1. Start Real Robot

Without driving the real robot, perform simulation demonstration of robotic arm motion in Movelt:

```
ros2 run dofbot_pro_driver dofbot_pro_driver
```

2.2. Launch Movelt2

```
ros2 launch dofbot_pro_moveit demo.launch.py
```

```

jetson@yahboom: ~
jetson@yahboom: ~ 132x19
[System Information]

ROS: humble
DOMAIN_ID: 25
IP_Address_1: 192.168.2.105
IP_Address_2: 172.17.0.1
jetson@yahboom:~$ ros2 run dofbot_pro_driver dofbot_pro_driver
[INFO] [1741861003.495313923] [joint_state_subscriber]: Subscribed to /joint_states
[INFO] [1741861013.856330900] [joint_state_subscriber]: Updated Joint Angles: [90.0, 90.0, 90.0, 90.0, 90.0, 30.0]

jetson@yahboom:~$ ros2 launch dofbot_pro_moveit demo.launch.py
[INFO] [launch]: All log files can be found below /home/jetson/.ros/log/2025-03-13-18-16-51-406301-yahboom-28321
[INFO] [launch]: Default logging verbosity is set to INFO
[INFO] [robot_state_publisher-1]: process started with pid [28398]
[INFO] [move_group-2]: process started with pid [28400]
[INFO] [rviz2-3]: process started with pid [28402]
[INFO] [ros2_control_node-4]: process started with pid [28404]
[INFO] [spawner-5]: process started with pid [28406]
[INFO] [spawner-6]: process started with pid [28408]
[INFO] [spawner-7]: process started with pid [28411]
[robot_state_publisher-1] [WARN] [1741861012.984268463] [kdl_parser]: The root link base_link has an inertia specified in the URDF, but KDL does not support a root link with an inertia. As a workaround, you can add an extra dummy link to your URDF.
[robot_state_publisher-1] [INFO] [1741861012.984529204] [robot_state_publisher]: got segment Arm1_Link
[robot_state_publisher-1] [INFO] [1741861012.984665046] [robot_state_publisher]: got segment Arm2_Link

```

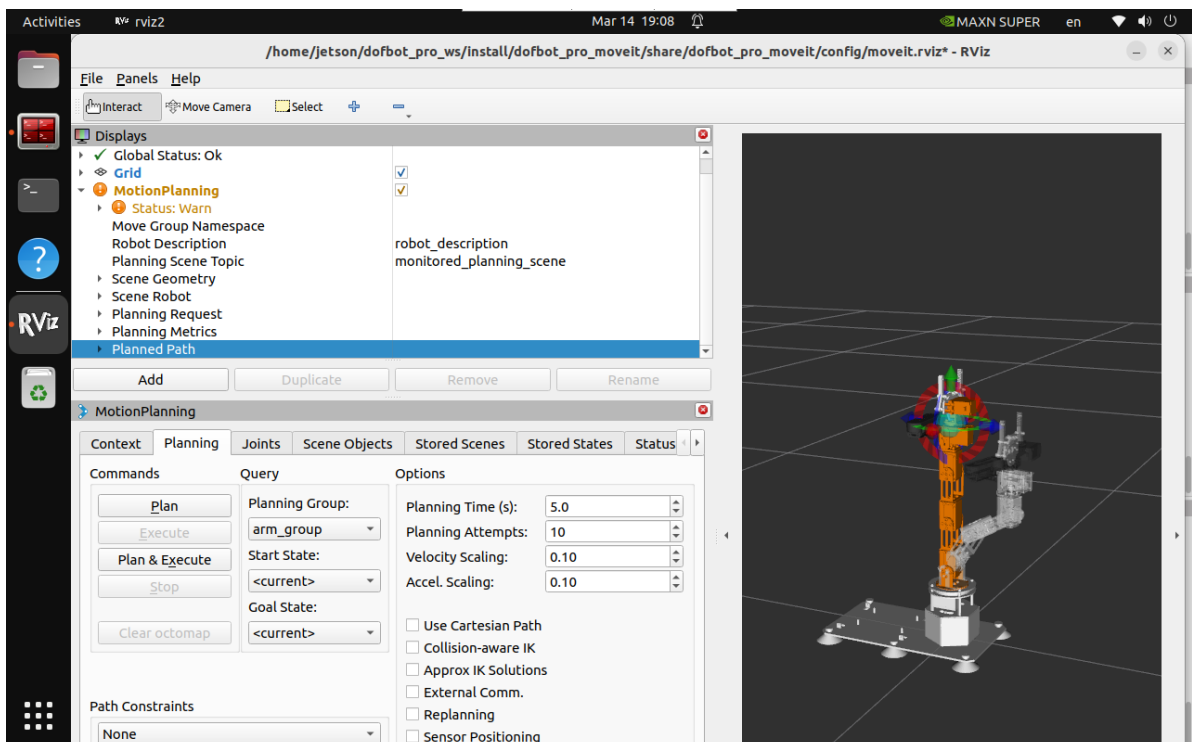
3. Inverse Kinematics Design

Inverse Kinematics refers to given the position of the robotic arm's end effector, MoveIt calculates the angles of each joint of the robotic arm, and then plans to the target pose by itself.

Launch Command

The robotic arm needs to be successfully loaded in MoveIt and `You can start planning now!` appears before running the following command: The robotic arm will plan to the position by itself

```
ros2 run dofbot_pro_moveit set_target_position
```



Precautions

Since it is difficult to determine the position of the robotic arm's end effector, it is recommended that users find reachable poses using the following method.

View the pose of `Arm5_Link` and then replace the target pose in the program: The robotic arm needs to be planned and executed to the specified pose for data changes to occur.

`MotionPlanning` → `Planned Path` → `Links`: Positions, Orientation

