MoveIt2 configuration

MoveIt2 is an advanced software framework for robot motion simulation, planning and control.

1. Environment setup

Motherboard: Jetson Orin Nano/Nx

ROS2: Humble

1.1. Install MoveIt2

```
sudo apt install ros-humble-moveit*
```

The Jetson Orin series motherboard can run MoveIt2 related cases directly on the motherboard, and the overall fluency is acceptable!

1.2. Create a folder

Create the configuration file automatically generated by MoveIt2:

```
mkdir ~/dofbot_pro_ws/src/dofbot_pro_moveit
```

2. MoveIt2 configuration

MoveIt Setup Assistant is a graphical tool to simplify and speed up the creation and configuration of MoveIt2 configuration packages.

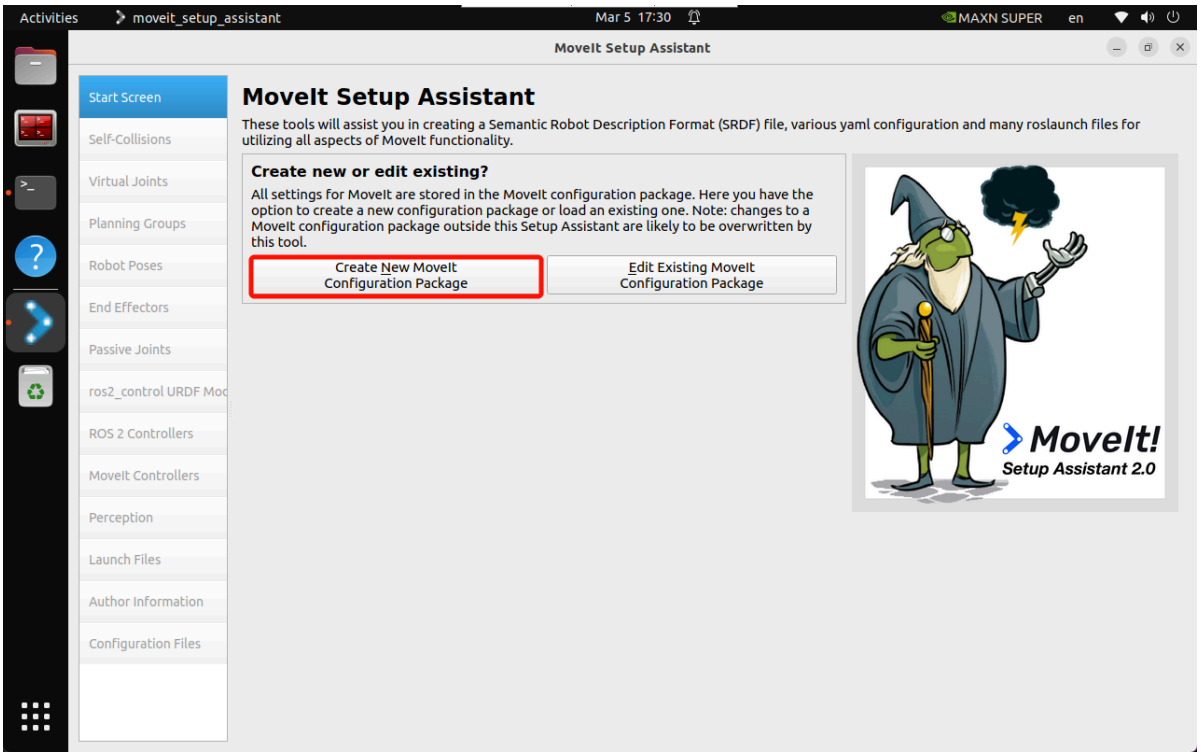
Through this tool, users can easily generate configuration files for robot motion planning and control without manually editing complex configuration files

2.1. Start the Assistant

Start MoveIt Setup Assistant:

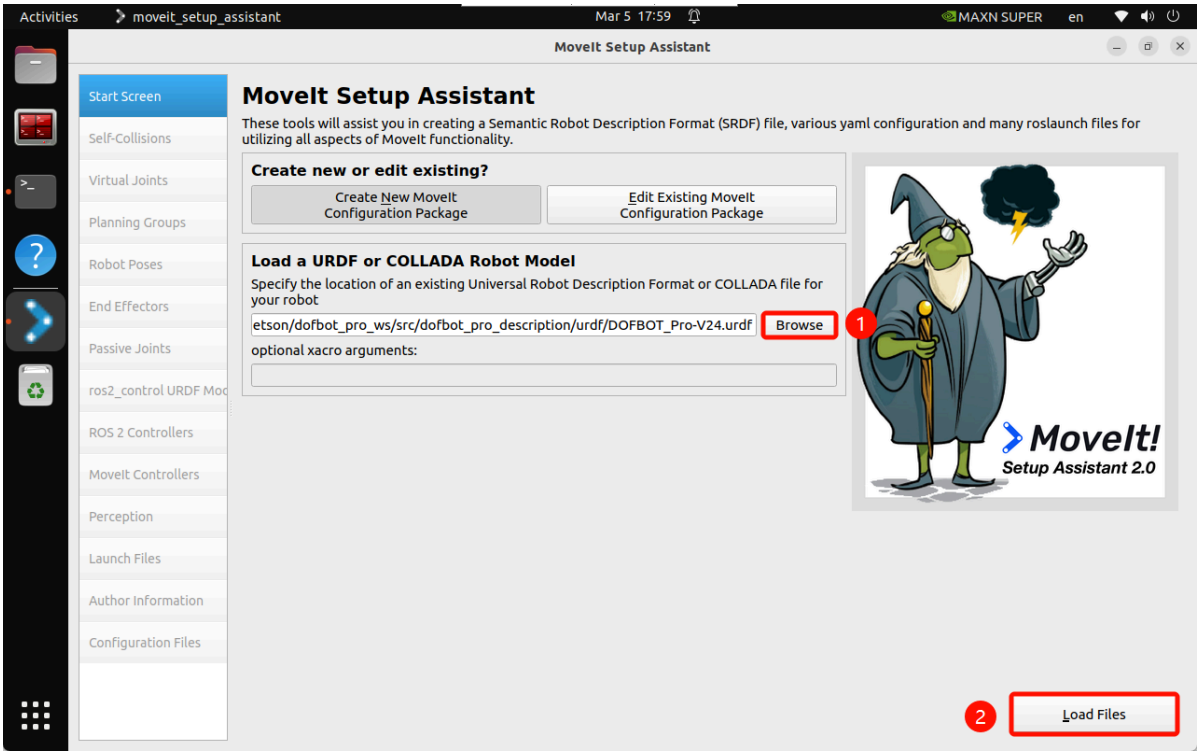
```
ros2 launch moveit_setup_assistant setup_assistant.launch.py
```

For the first time, you can choose to create a new MoveIt configuration package. For simple modifications later, you can choose to compile the existing MoveIt package:

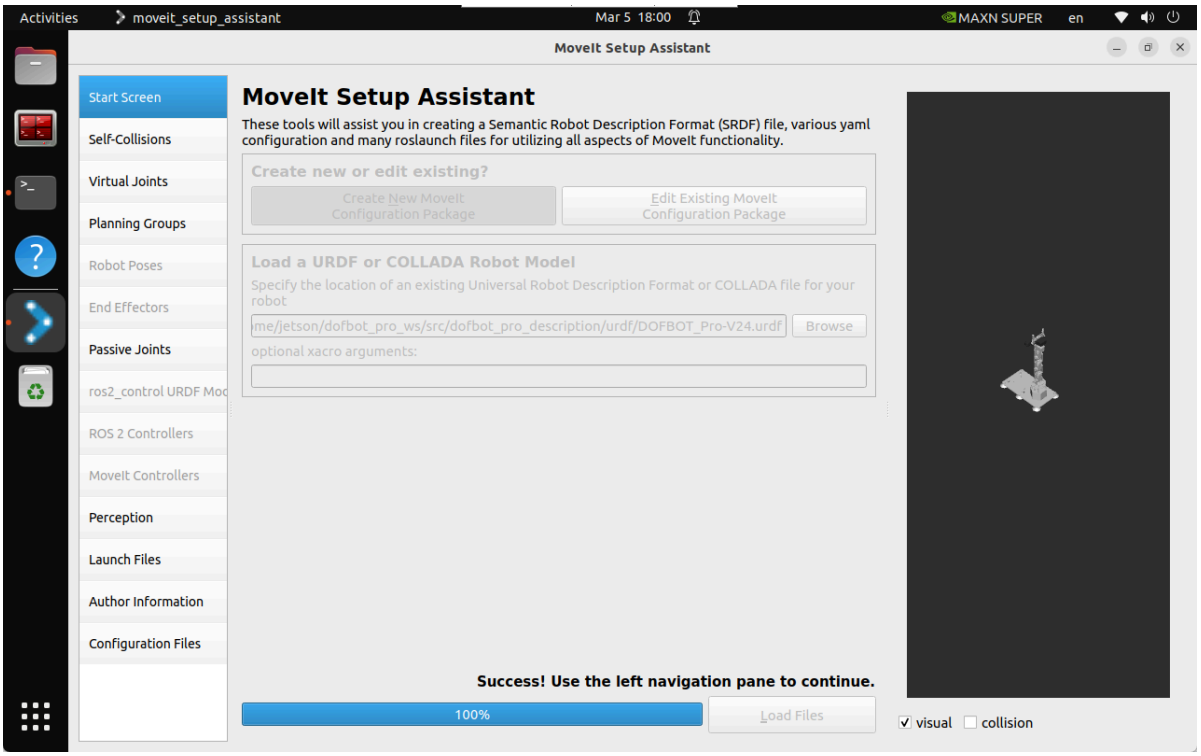


To create a new MoveIt configuration package, you need to import the robot's URDF model: Simply importing the URDF model file into the MoveIt Assistant will report an error, and you need to use the compiled package (dofbot_pro_description) URDF file path

```
/home/jetson/dofbot_pro_ws/src/dofbot_pro_description/urdf/DOFBOT_Pro-V24.urdf
```



After selecting the URDF file, click Load File and the MoveIt Assistant will display the robot model:

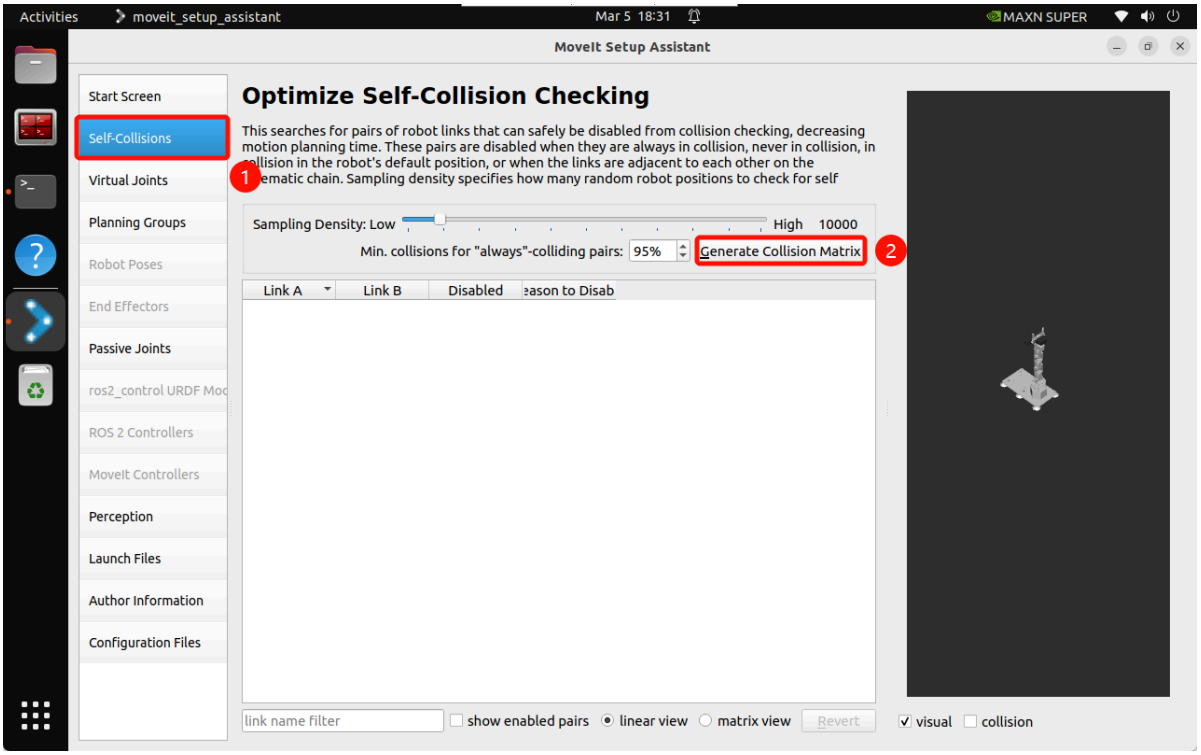


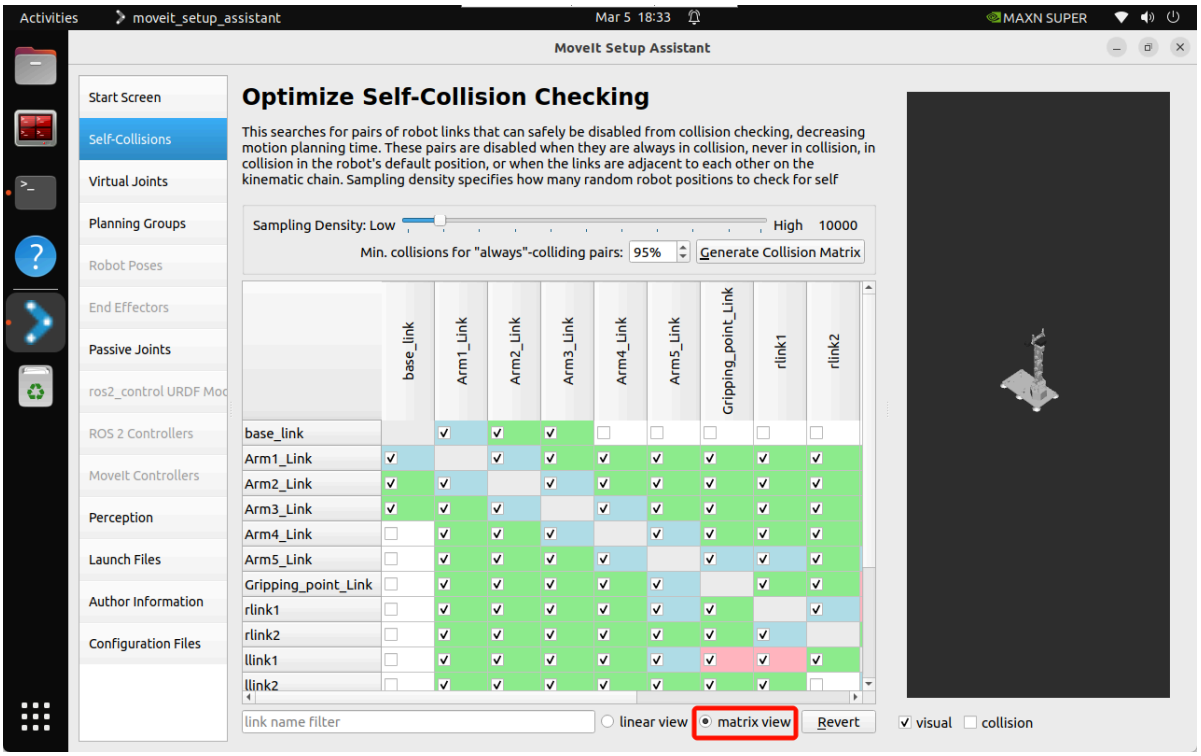
2.2, Collision Detection

The self-collision matrix is a function used to optimize motion planning. Its main function is to generate a matrix to describe whether collisions may occur between the links in the robot model.

This matrix can help the motion planner avoid unnecessary collision detection when planning the path, thereby improving planning efficiency.

To generate a self-collision matrix, you can click the MoveIt Setup Assistant option to automatically generate it:

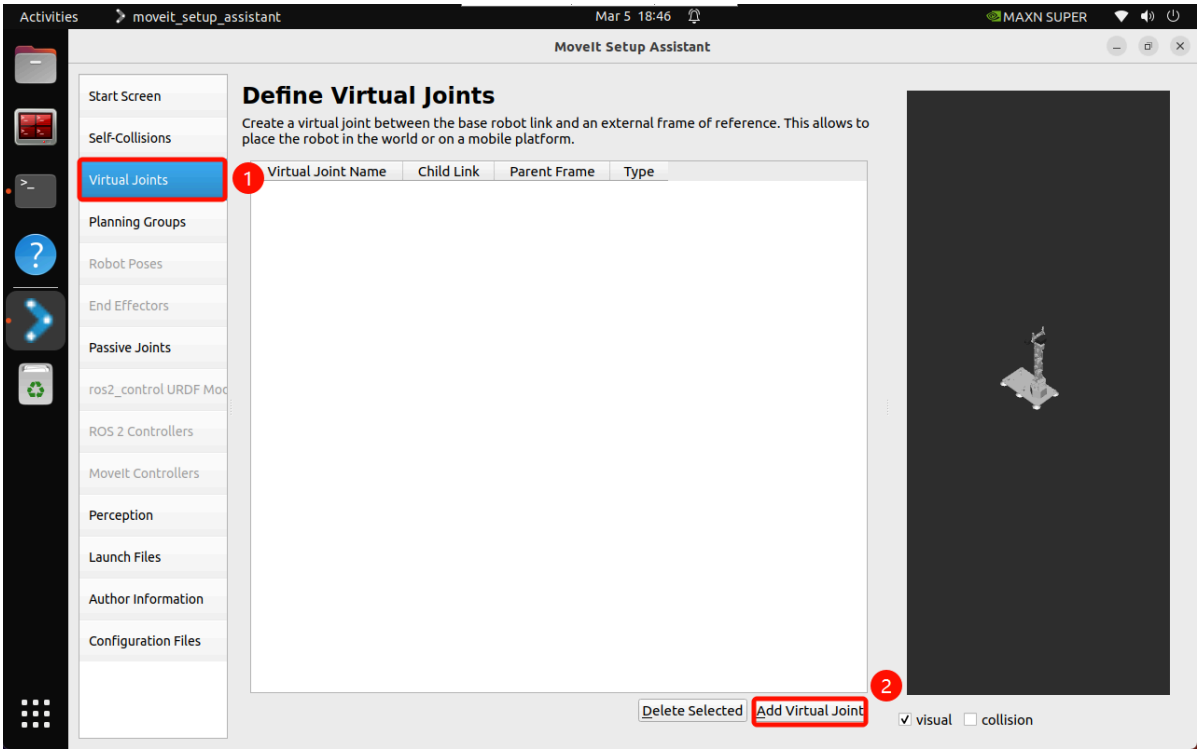




2.3. Virtual joints

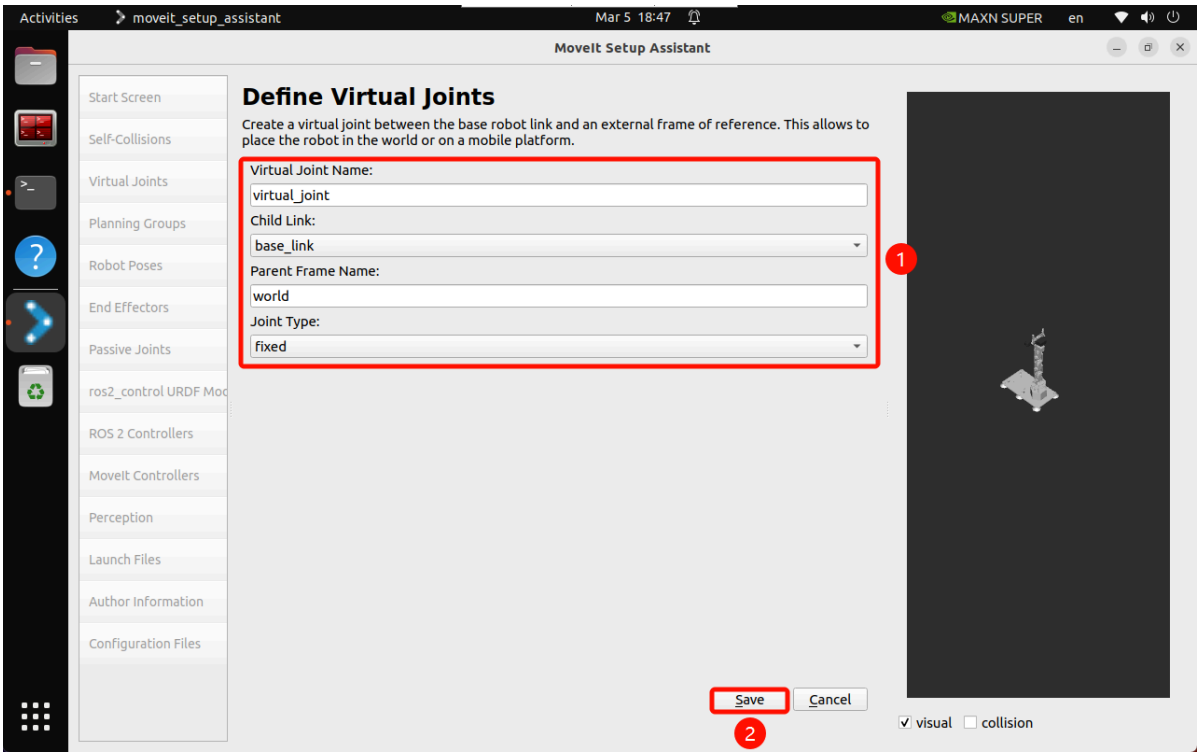
A virtual joint is used to define the relationship between the robot model and the external world.

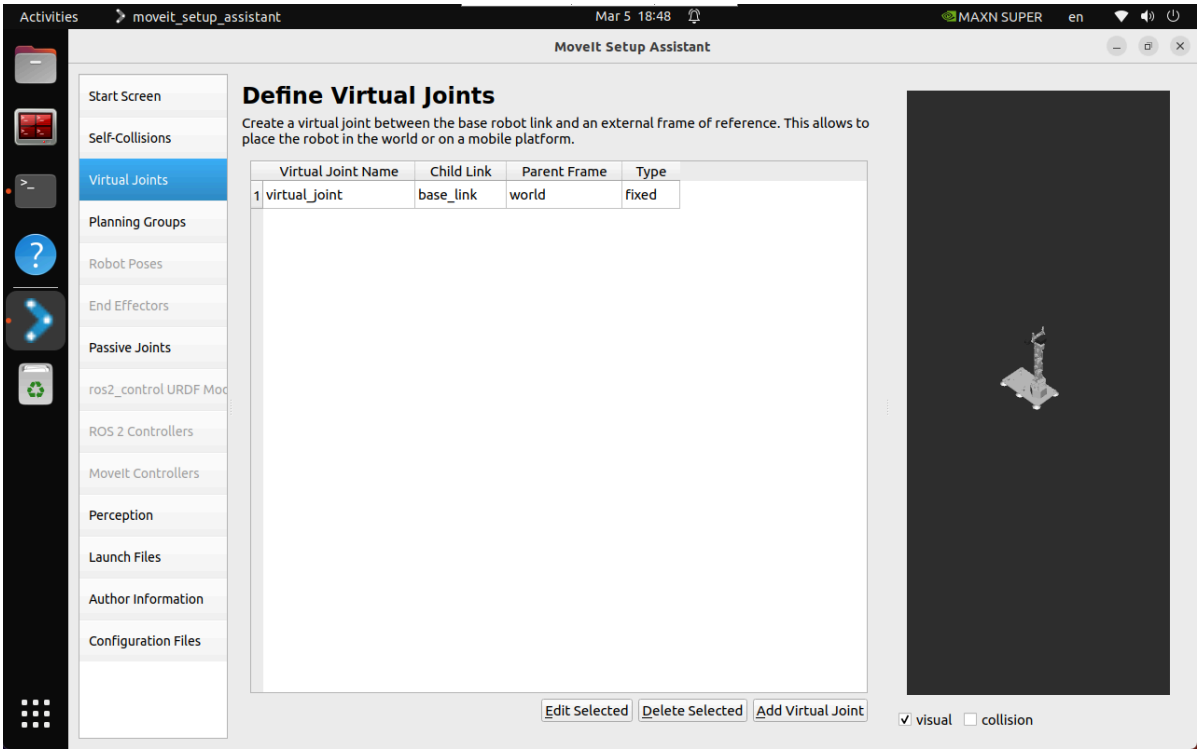
Virtual joints are usually used to describe the connection between the robot base and a fixed reference system (such as the world coordinate system).



Fix the robot base link `base_link` and `world`:

- Virtual Joint Name: virtual_joint
- Child Link: base_link
- Parent Frame Name: world
- Joint Type: fixed

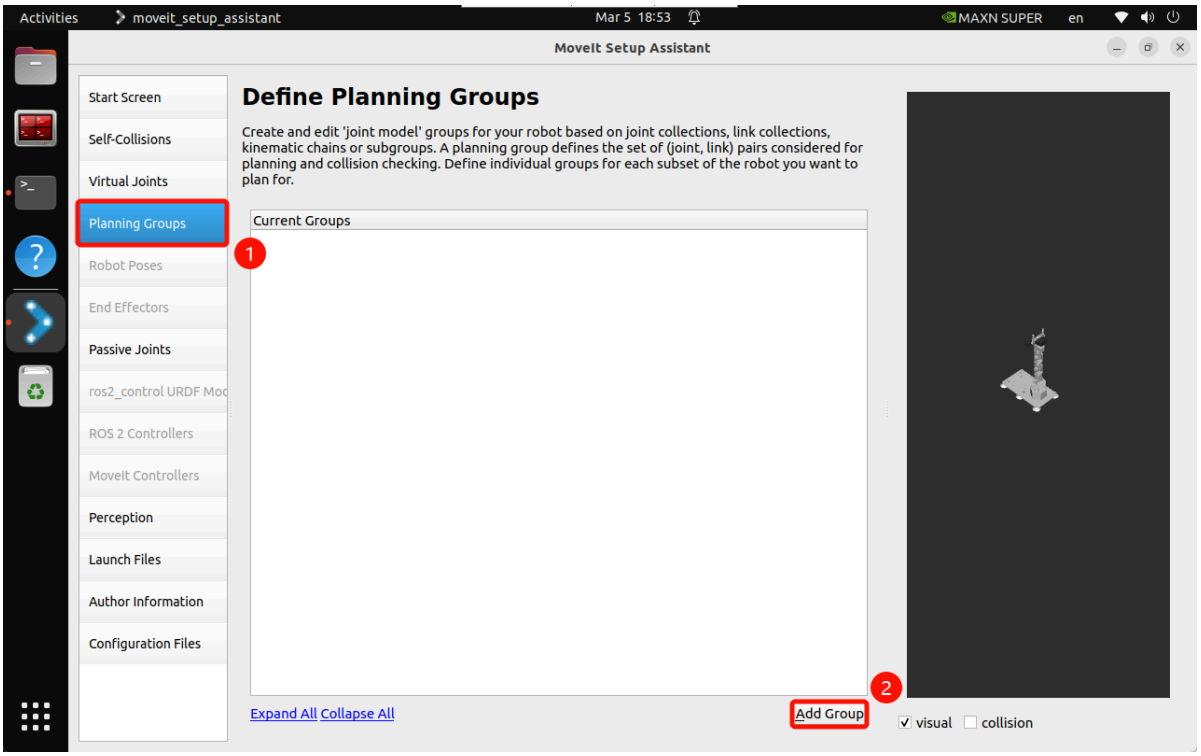




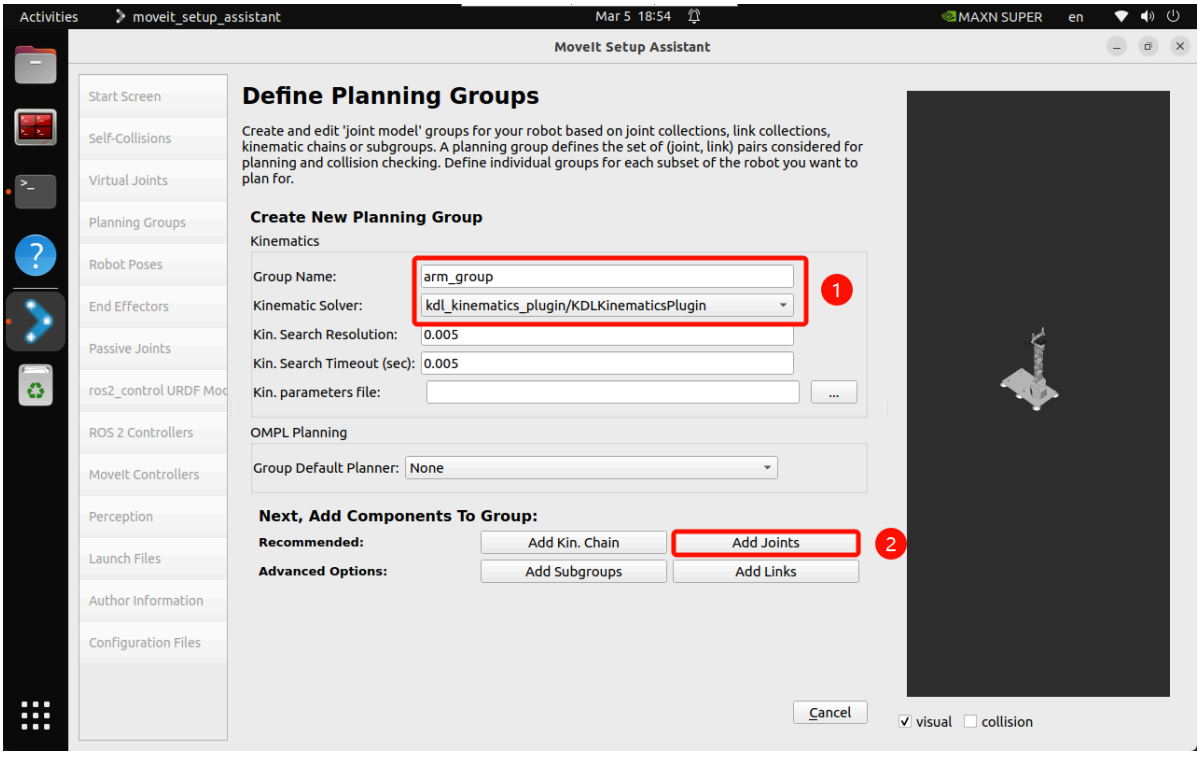
2.4, Motion Planning Group

The planning group is a key step in configuring the robot's motion planning.

The planning group defines which joints and links in the robot can move together and how to plan their motion.

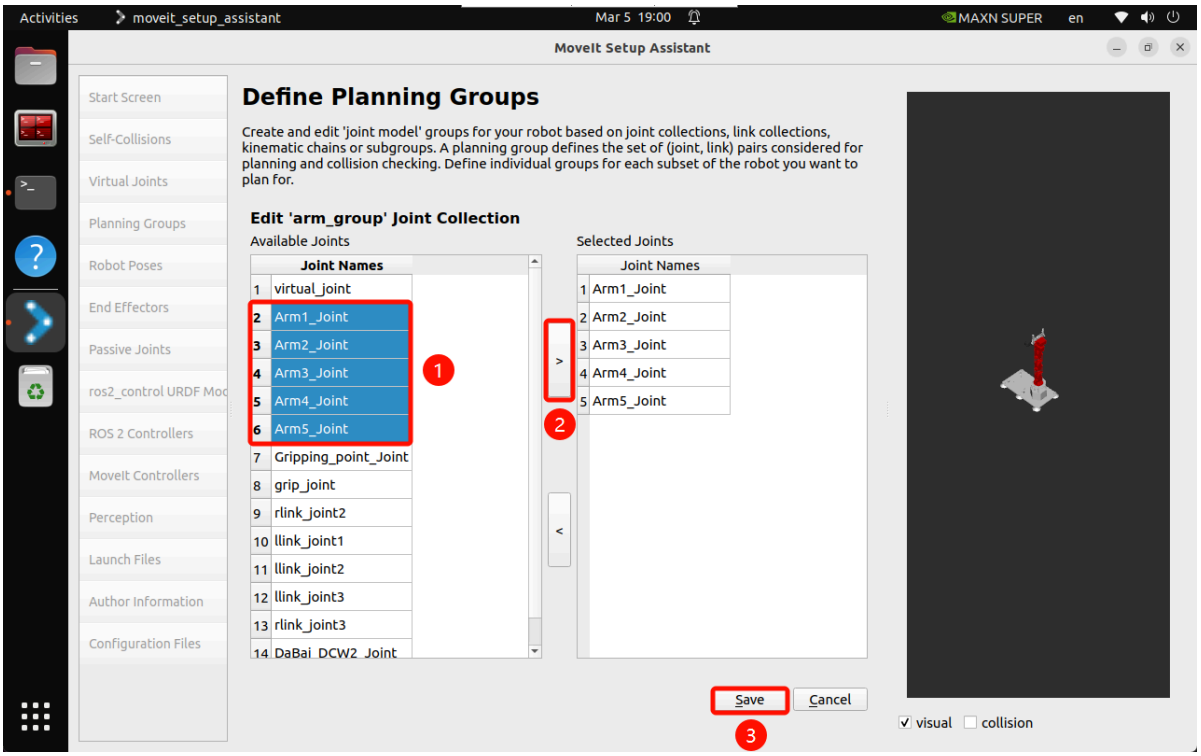


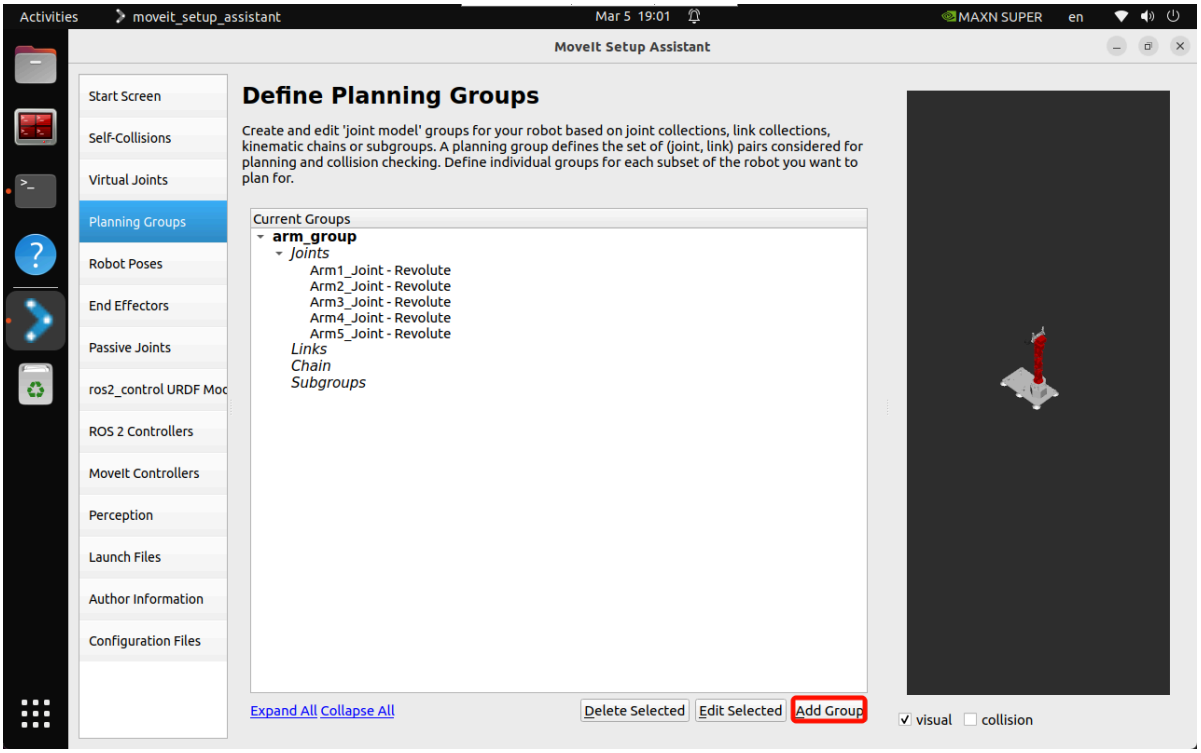
- Planning robot arm group: arm_group
  - Group Name: arm_group
  - Kinematic Solver: kdl_kinematics_plugin/KDLKinematicsPlugin
  - Kin. Search Resolution (sampling density of joint space): 0.005
  - Kin. Search Timeout (solution time): 0.005



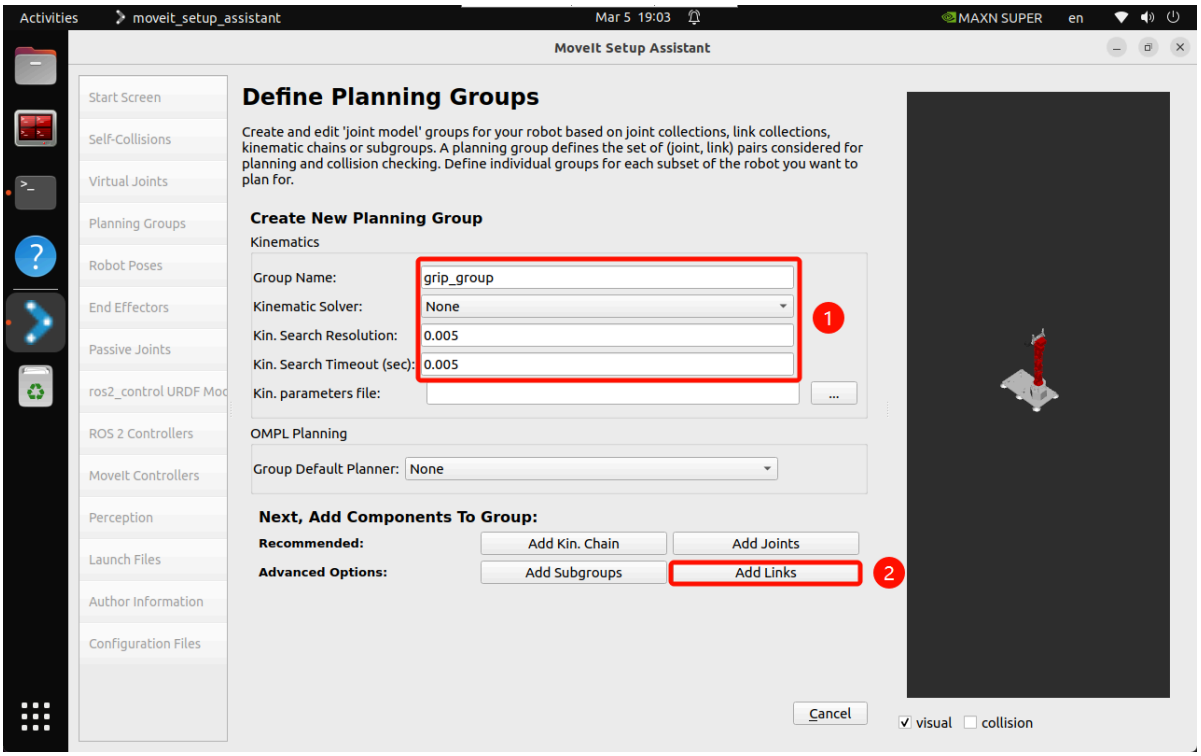
- Add Joints: Select the joints of the robot arm

Joints：Arm1_Joint、Arm2_Joint、Arm3_Joint、Arm4_Joint、Arm5_Joint



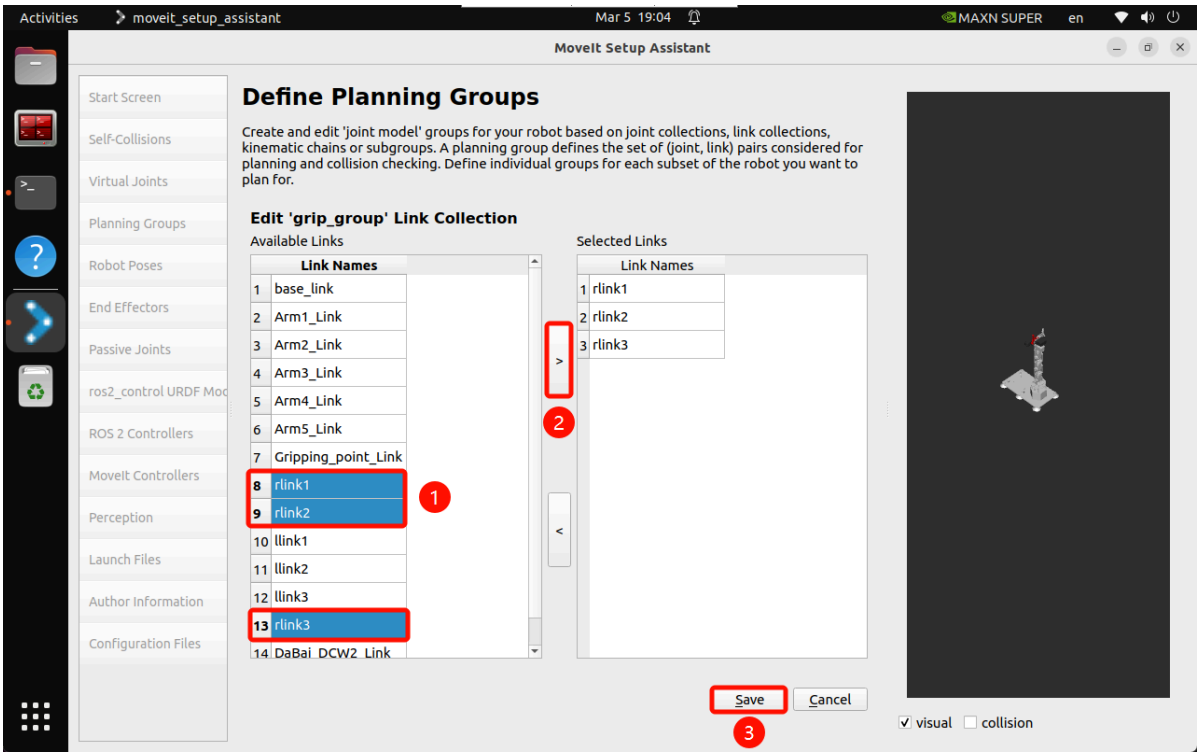


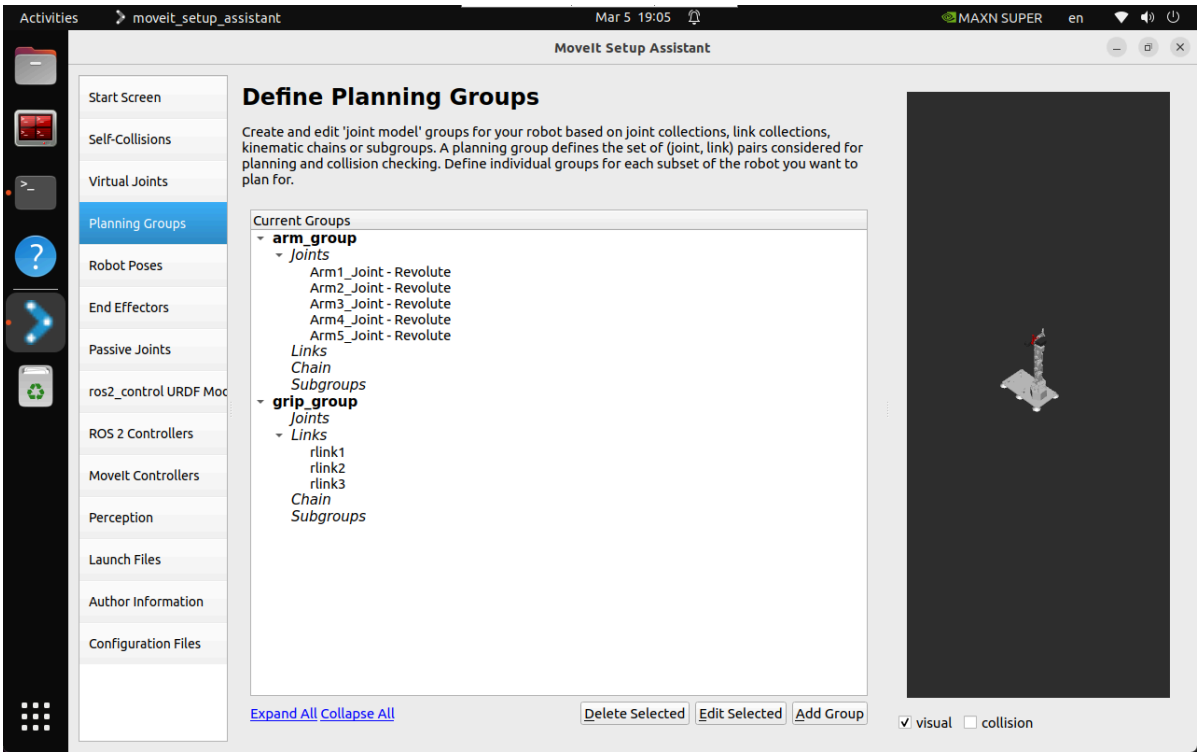
- Planning gripper group: grip_group
  - Group Name: grip_group
  - Kinematic Solver: None
  - Kin. Search Resolution: 0.005
  - Kin. Search Timeout: 0.005



- Add Links: Select the connecting rod of the gripper

Links：rlink1、rlink2、rlink3

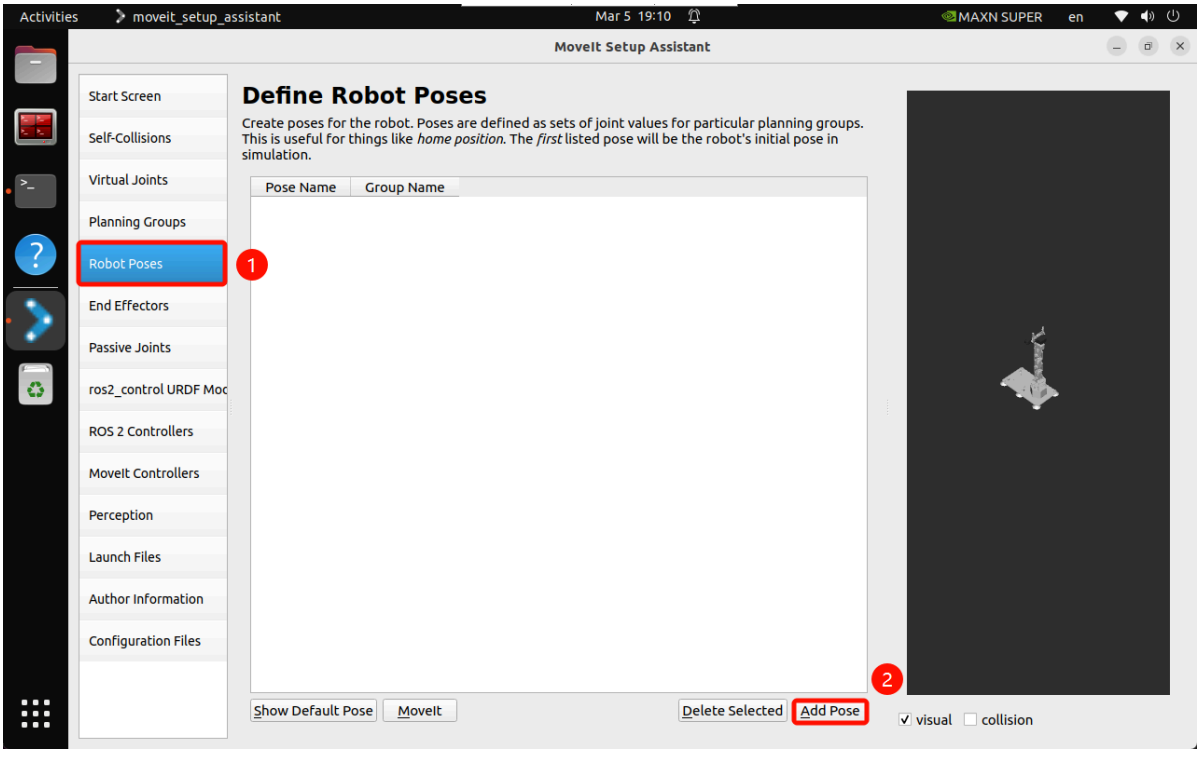




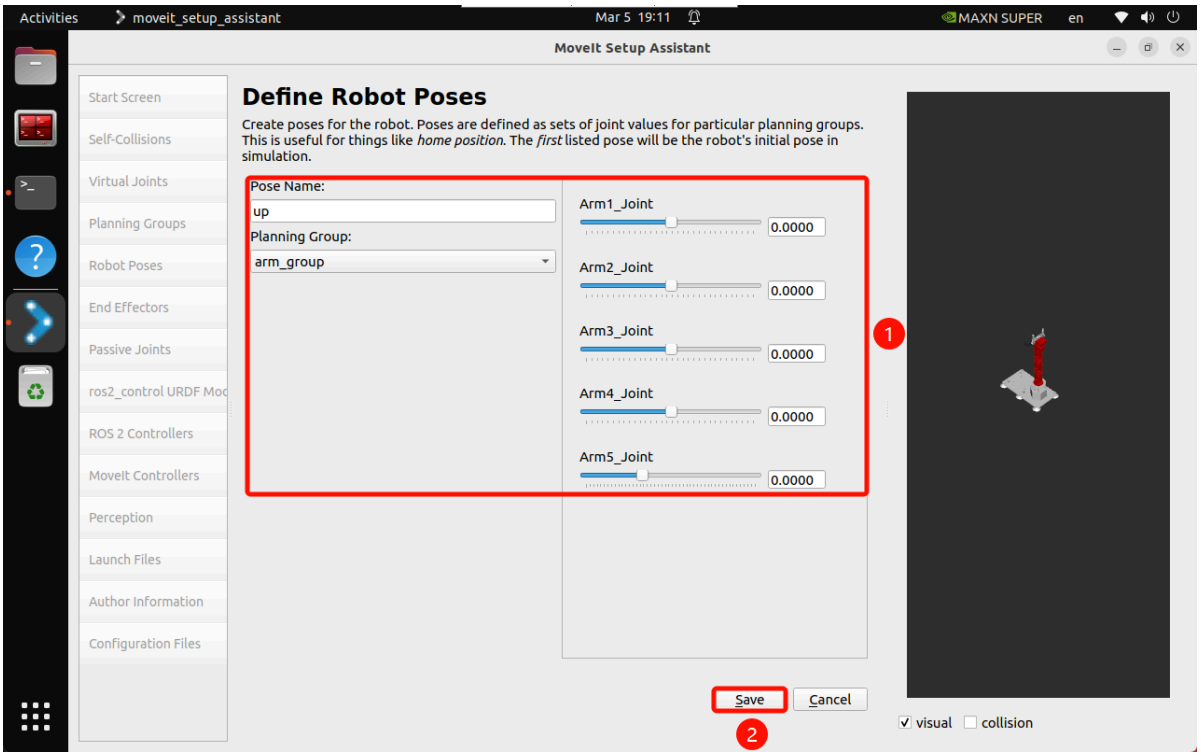
2.5. Robot position

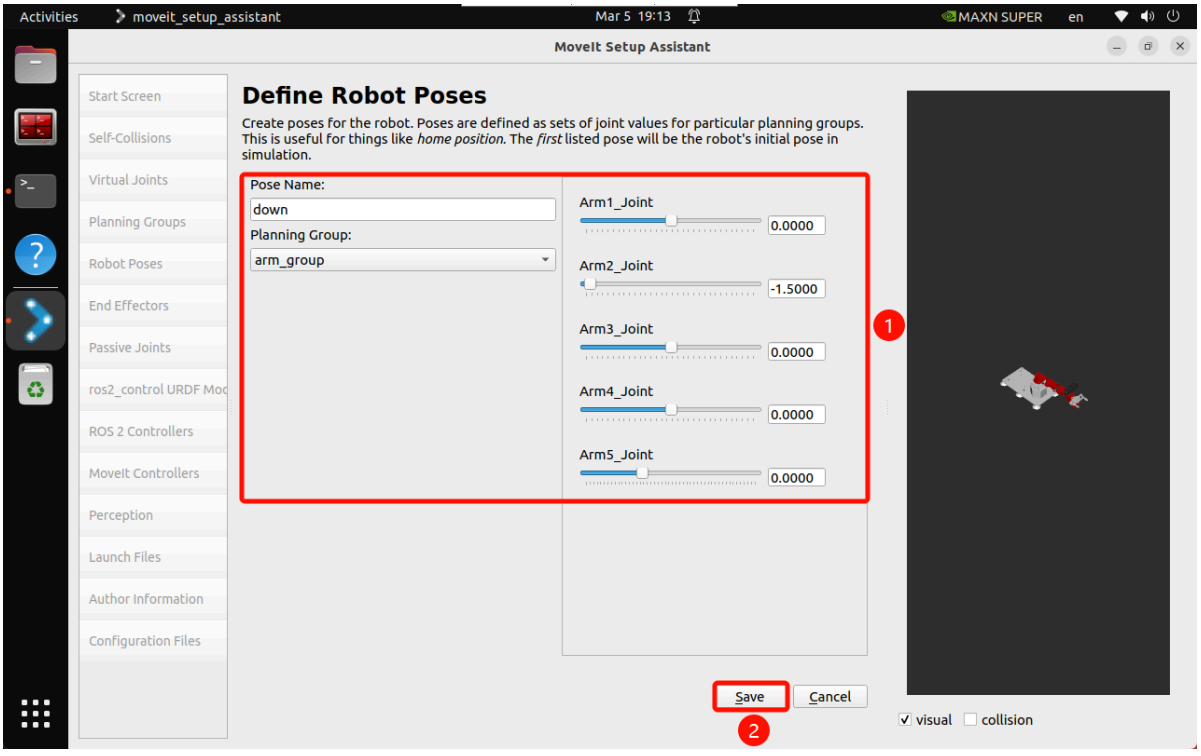
> ### Robot arm position

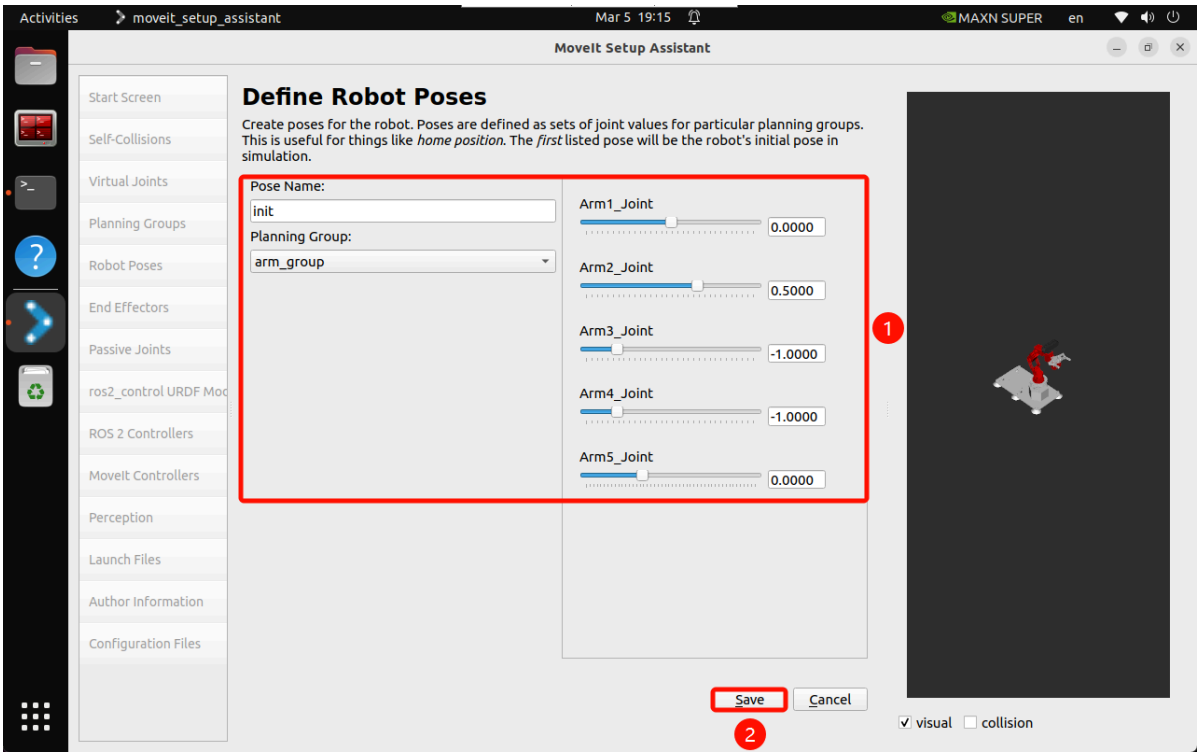
Set three sets of robot arm predefined postures to the robot configuration:



You can manually move the slider to set the state of each joint: up, down, init

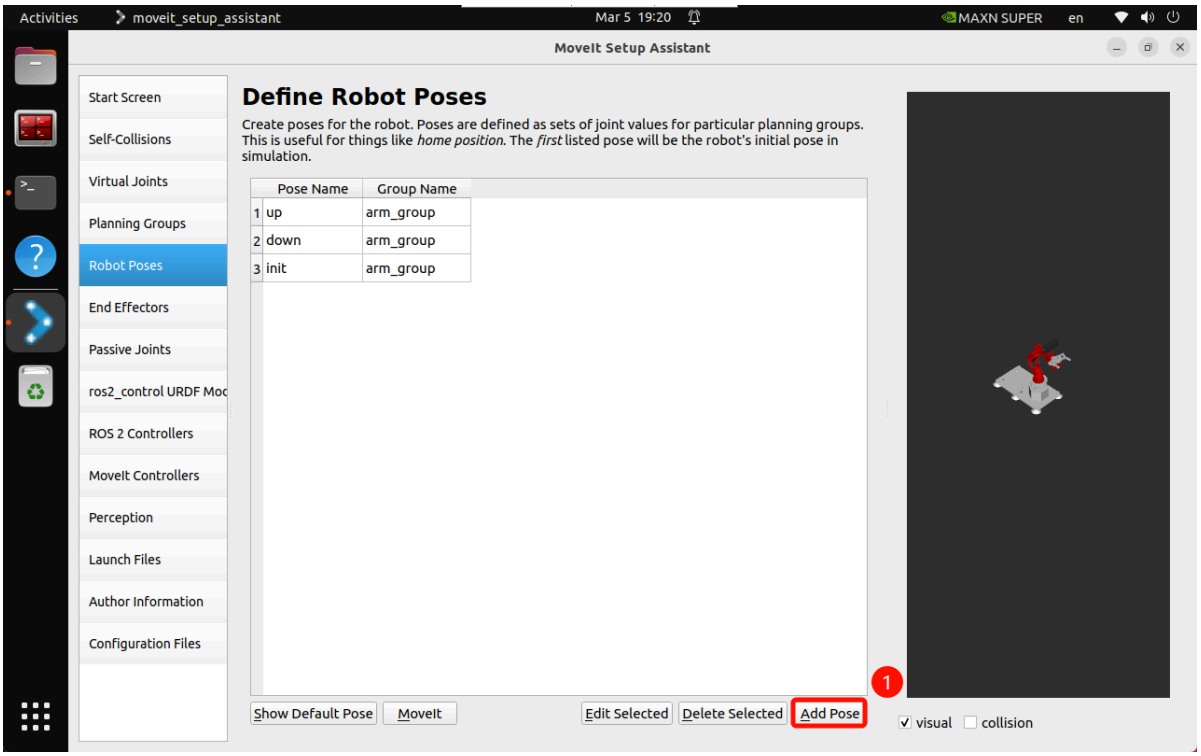

Activities
moveit_setup_assistant
Mar 5 19:11
MAXN SUPER
en
MoveIt Setup Assistant
Start Screen
Self-Collisions
Virtual Joints
Planning Groups
Robot Poses
End Effectors
Passive Joints
ros2_control URDF Mod
ROS 2 Controllers
MoveIt Controllers
Perception
Launch Files
Author Information
Configuration Files
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 listed pose will be the robot's initial pose in simulation.
Pose Name:
up
Planning Group:
arm_group
Arm1_Joint
0.0000
Arm2_Joint
0.0000
Arm3_Joint
0.0000
Arm4_Joint
0.0000
Arm5_Joint
0.0000
1
Save
Cancel
2
visual
collision


Activities
moveit_setup_assistant
Mar 5 19:13
MAXN SUPER
en
MoveIt Setup Assistant
Start Screen
Self-Collisions
Virtual Joints
Planning Groups
Robot Poses
End Effectors
Passive Joints
ros2_control URDF Mod
ROS 2 Controllers
MoveIt Controllers
Perception
Launch Files
Author Information
Configuration Files
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 listed pose will be the robot's initial pose in simulation.
Pose Name:
down
Planning Group:
arm_group
Arm1_Joint
0.0000
Arm2_Joint
-1.5000
Arm3_Joint
0.0000
Arm4_Joint
0.0000
Arm5_Joint
0.0000
1
Save
Cancel
2
visual
collision

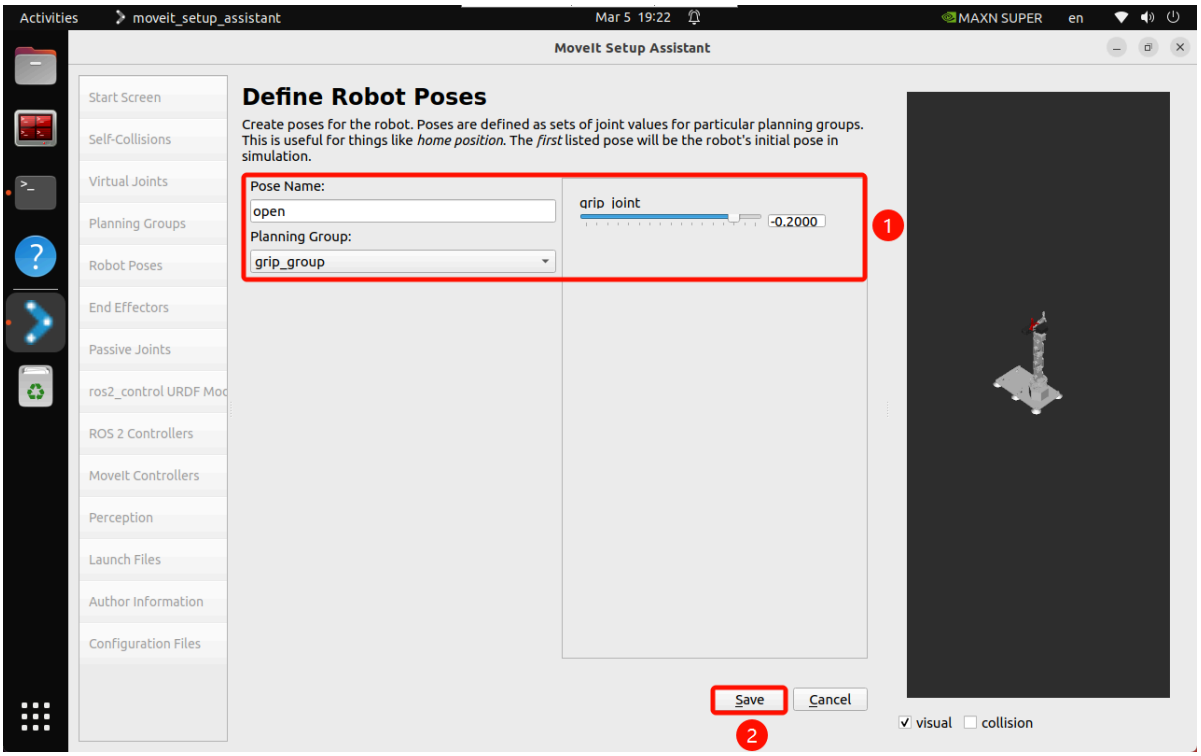


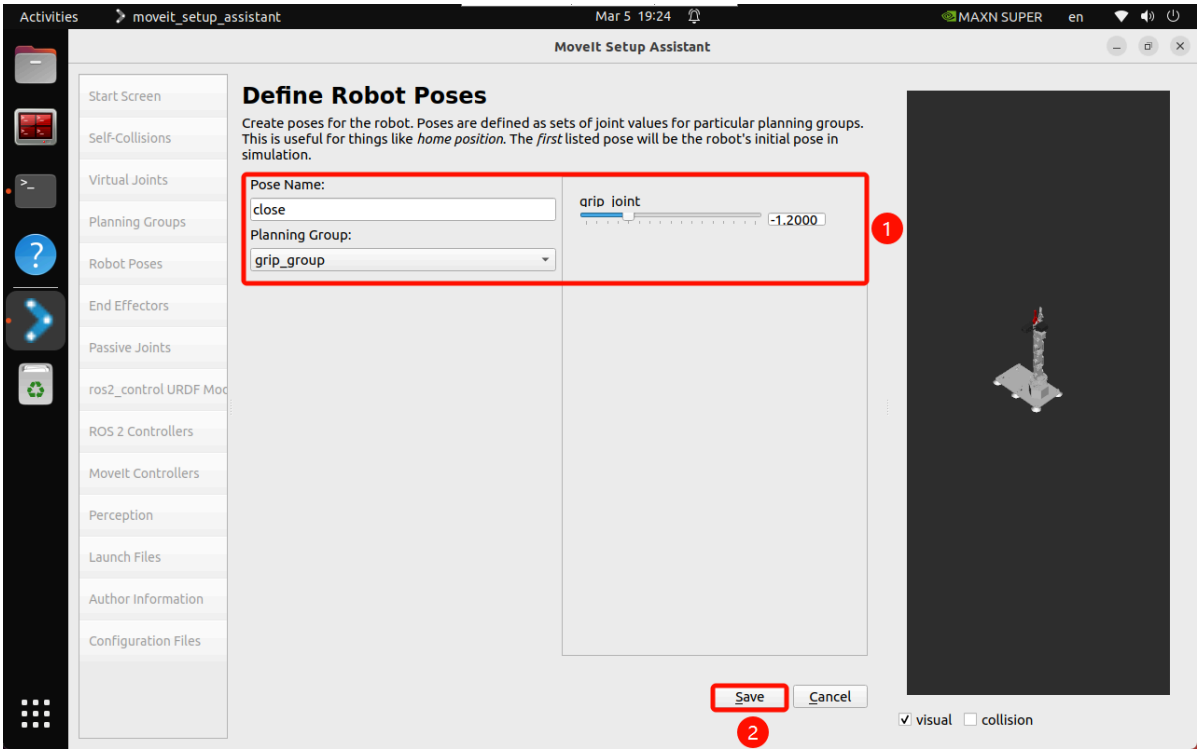
Jaw Position

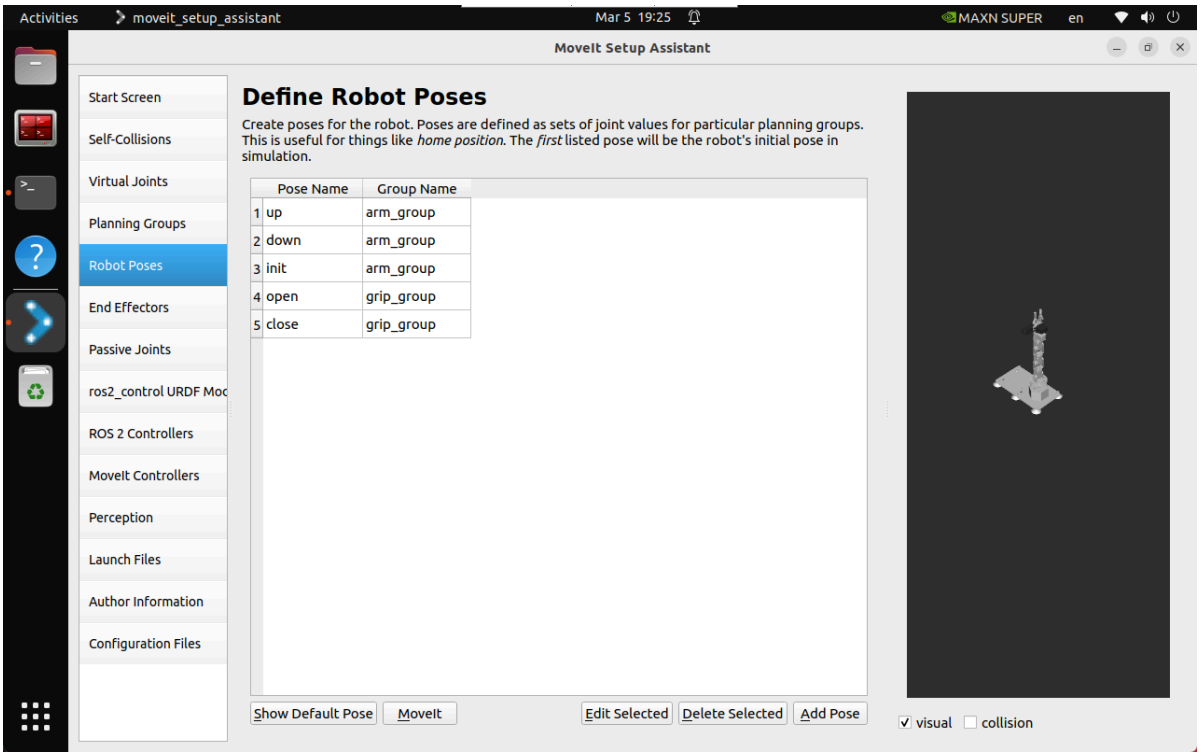
Set two sets of predefined gripper poses to the robot configuration:



You can manually move the slider to set the gripper state: open, close

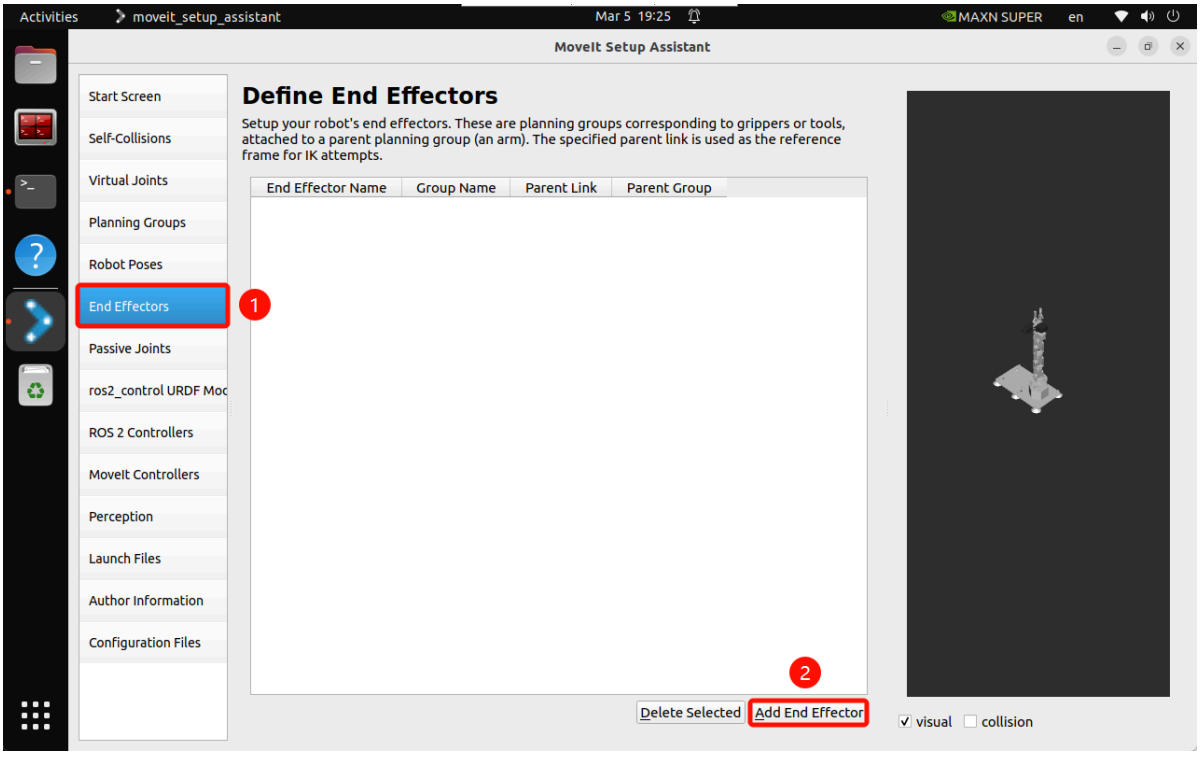

Activities
moveit_setup_assistant
Mar 5 19:22
MAXN SUPER
en
MoveIt Setup Assistant
Start Screen
Self-Collisions
Virtual Joints
Planning Groups
Robot Poses
End Effectors
Passive Joints
ros2_control URDF Mod
ROS 2 Controllers
MoveIt Controllers
Perception
Launch Files
Author Information
Configuration Files
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 listed pose will be the robot's initial pose in simulation.
Pose Name:
open
Planning Group:
grip_group
grip_joint
-0.2000
1
Save
Cancel
2
visual
collision


Activities
moveit_setup_assistant
Mar 5 19:24
MAXN SUPER
en
MoveIt Setup Assistant
Start Screen
Self-Collisions
Virtual Joints
Planning Groups
Robot Poses
End Effectors
Passive Joints
ros2_control URDF Mod
ROS 2 Controllers
MoveIt Controllers
Perception
Launch Files
Author Information
Configuration Files
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 listed pose will be the robot's initial pose in simulation.
Pose Name:
close
Planning Group:
grip_group
grip_joint
-1.2000
1
Save
Cancel
2
visual
collision



2.6. End effector

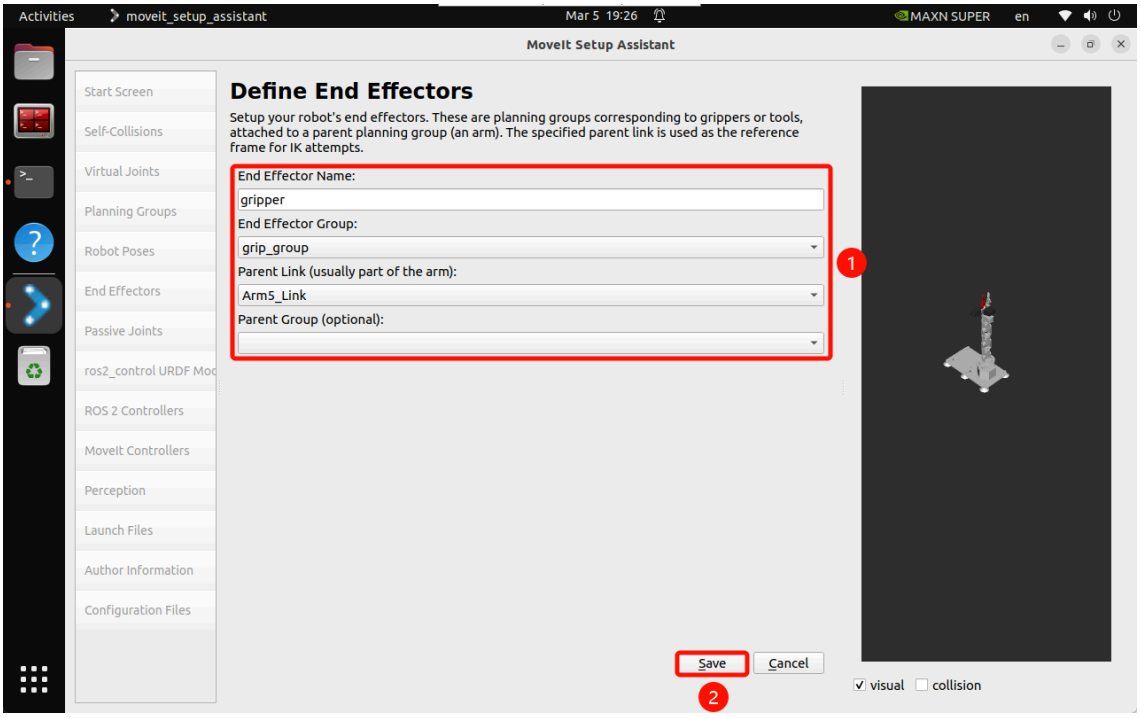
Set the gripper group as the end effector:

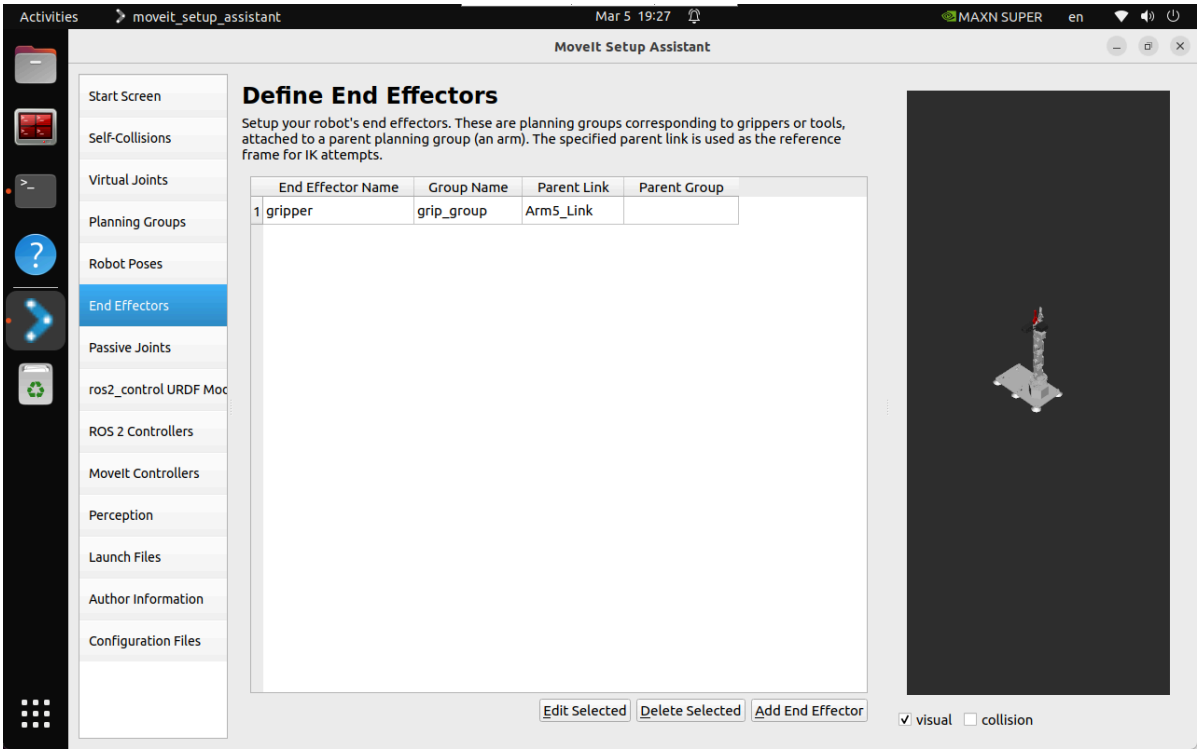


End Effector: gripper

- End Effector Name: gripper
- End Effector Group: gripp_group
- Parent Link: Arm5_Link

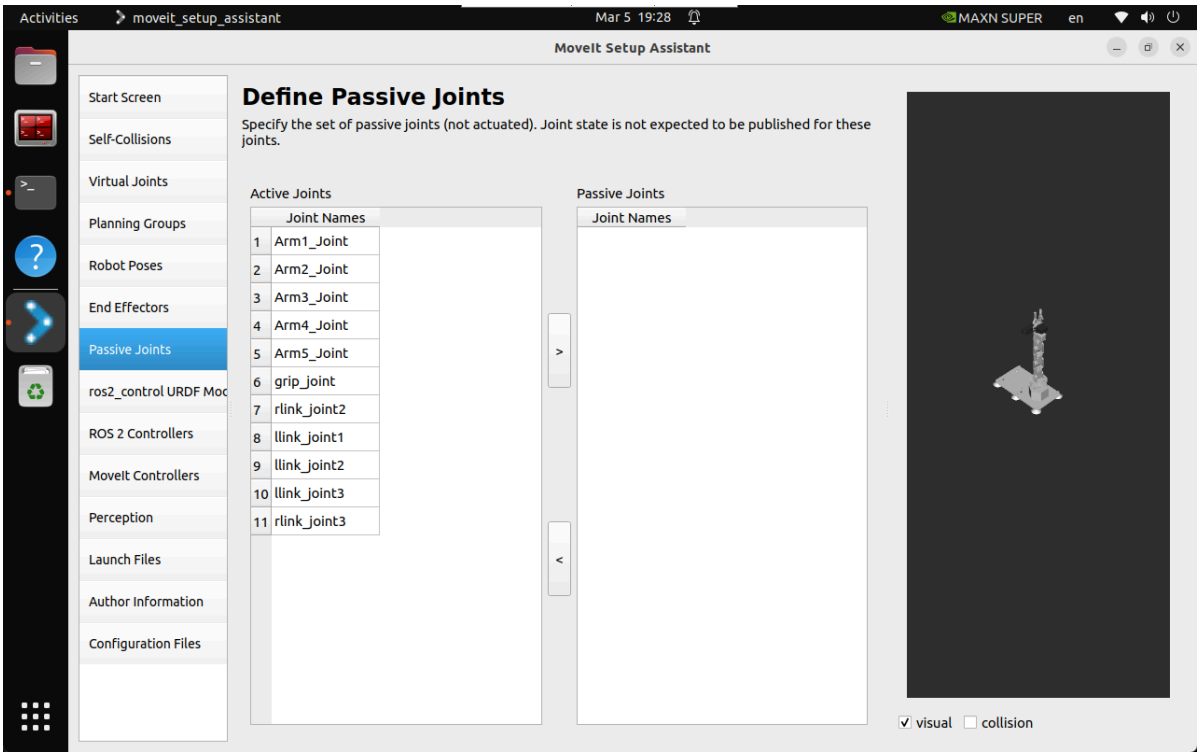
- Parent Group: None





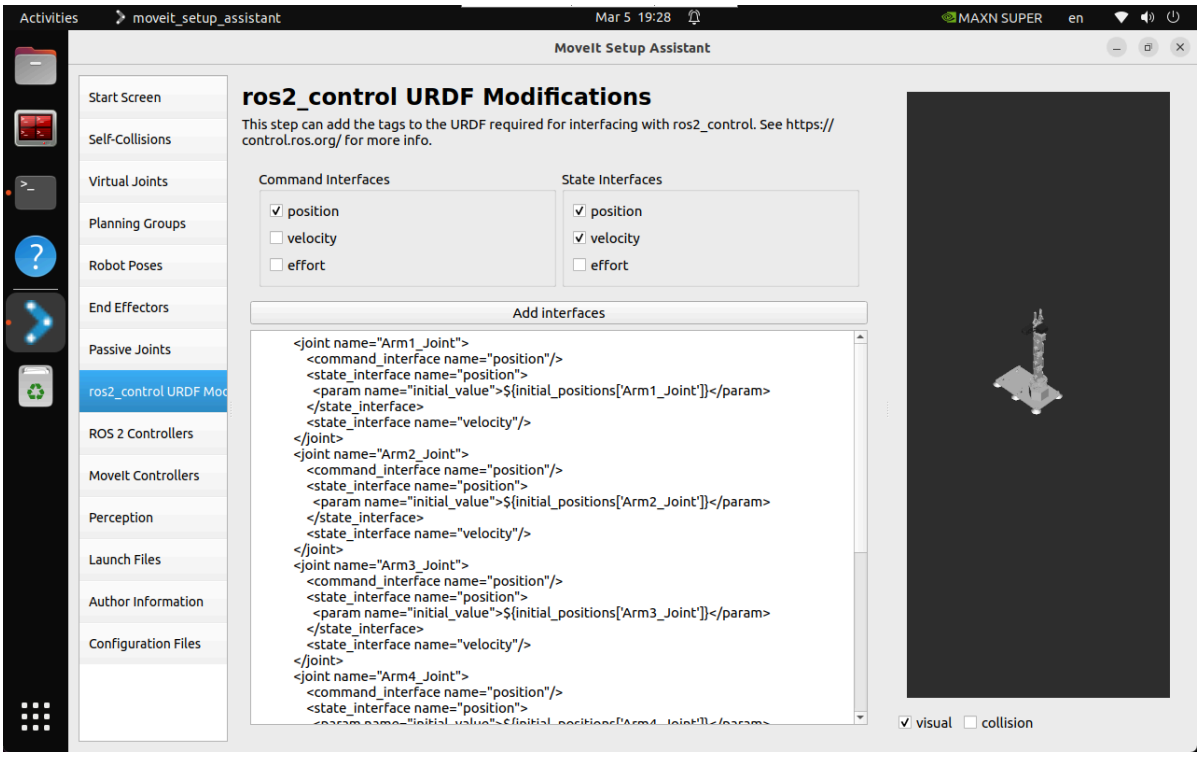
2.7 Passive joints

Passive joints are non-actuated joints that cannot be directly controlled: if the robot arm does not have passive joints, it will be skipped directly.



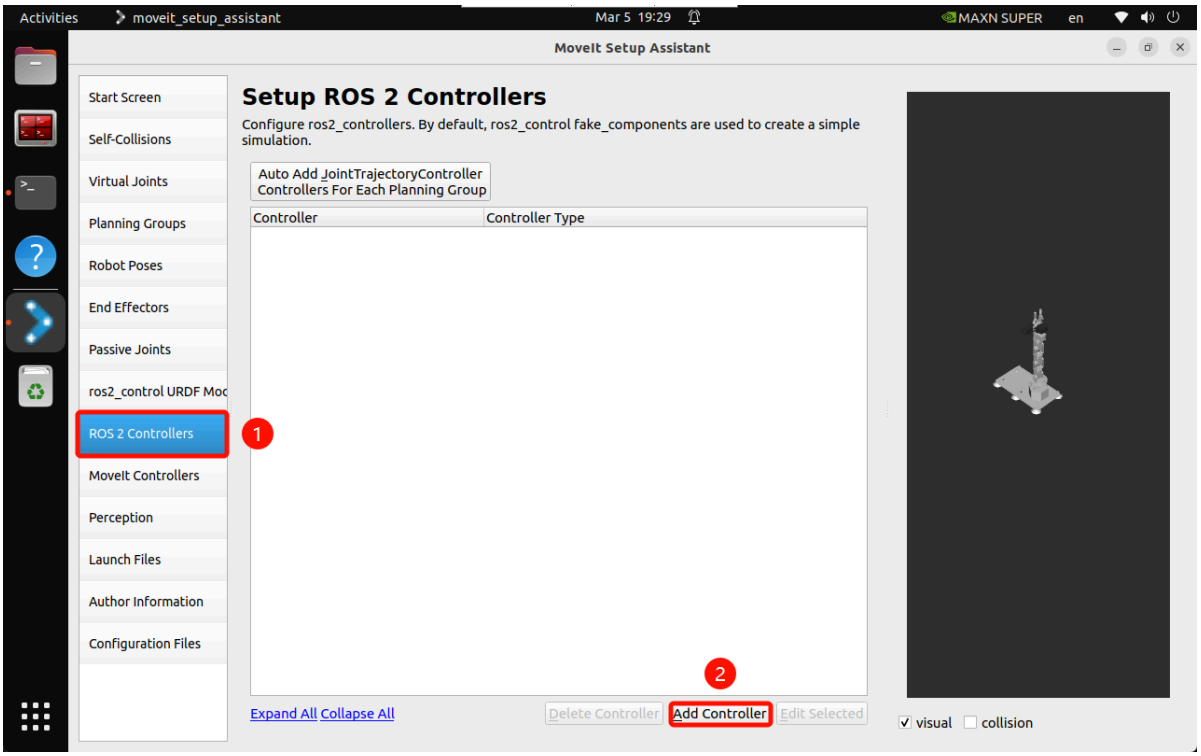
2.8, URDF file

MoveIt Setup Assistant will automatically set up the command interface and status interface for each joint: use the default generated information.



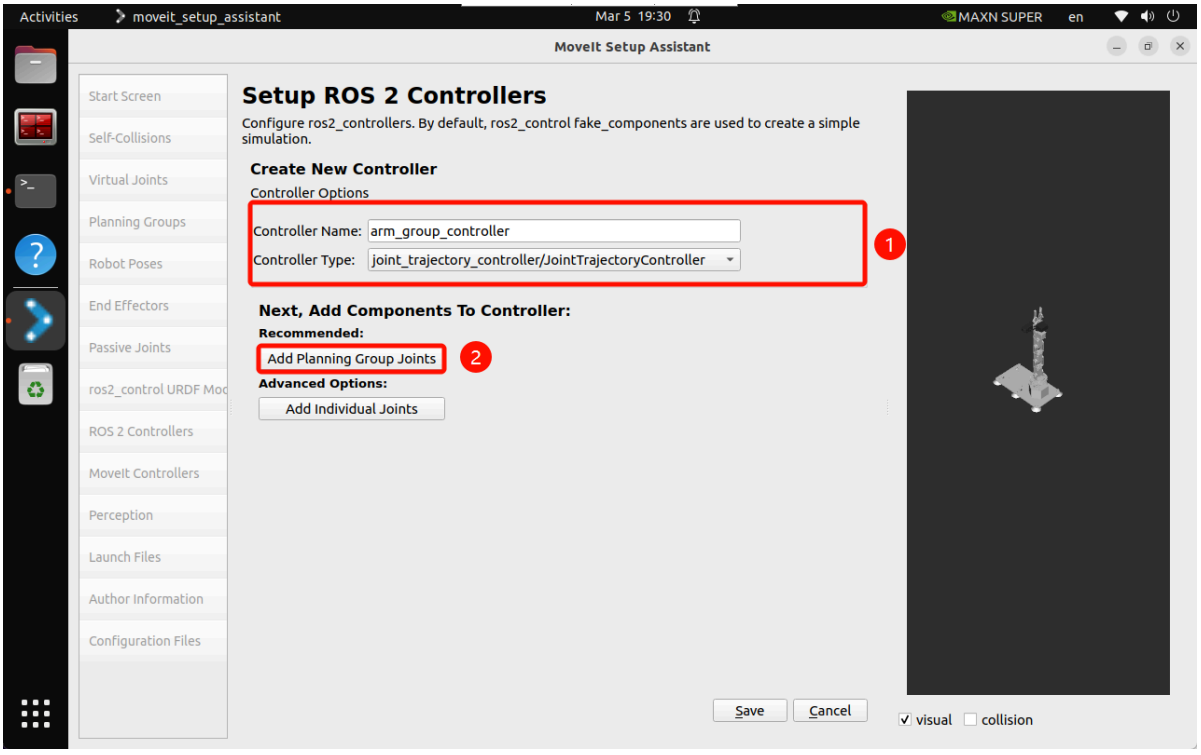
2.9, ROS2 Controllers

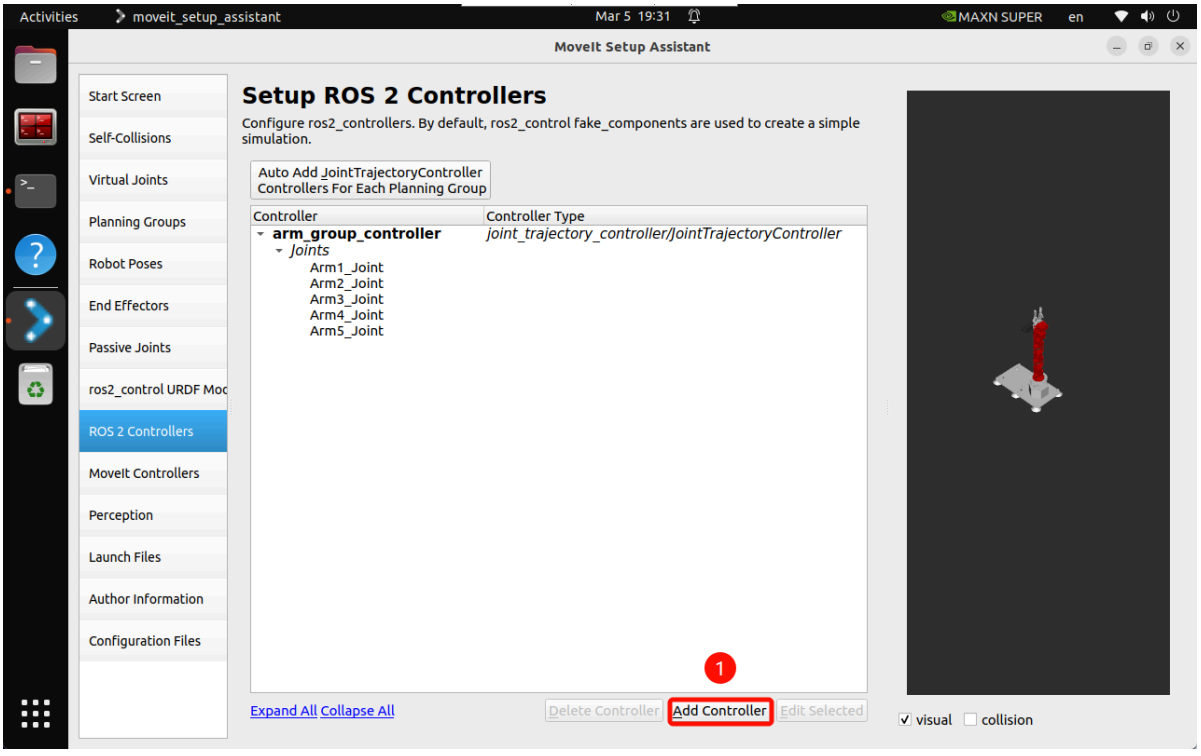
ROS2 Controllers is a framework for real-time robot control, which can be used to automatically generate simulation controllers to drive robot joints.



Add a robotic arm controller:

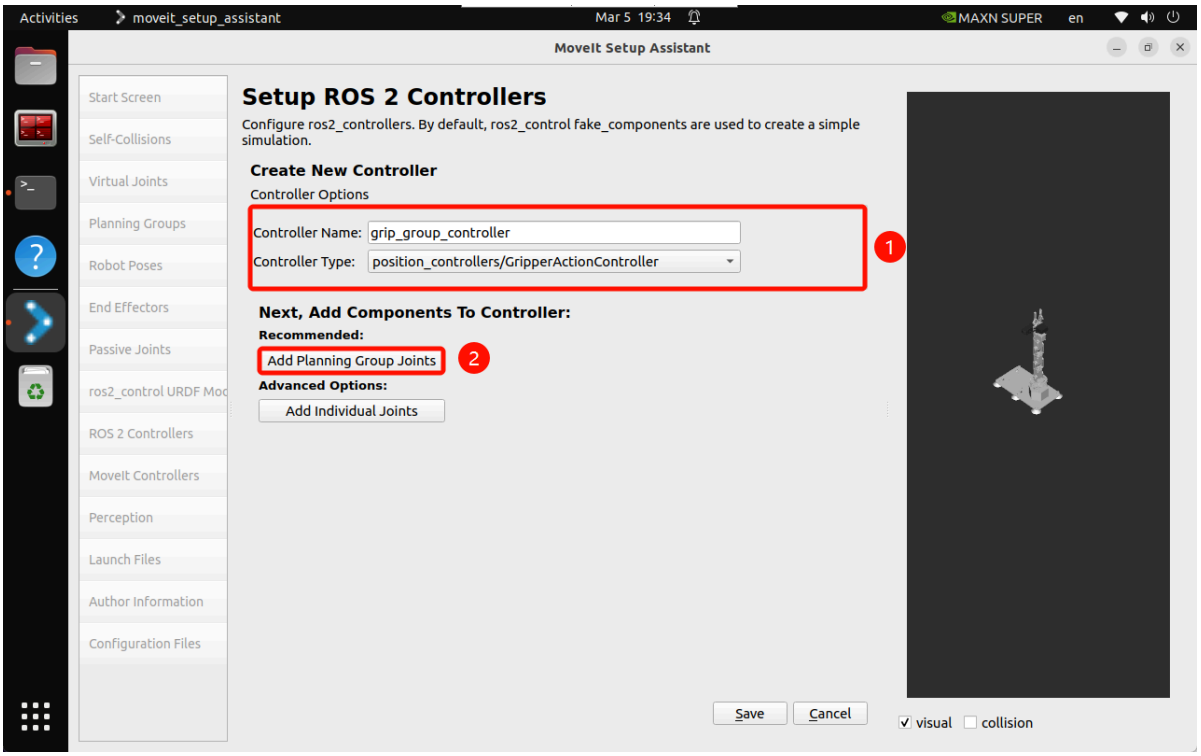
- Controller Name：arm_group_controller
- Controller Type：joint_trajectory_controller/JointTrajectoryController
- Add Planning Group Joints：arm_group



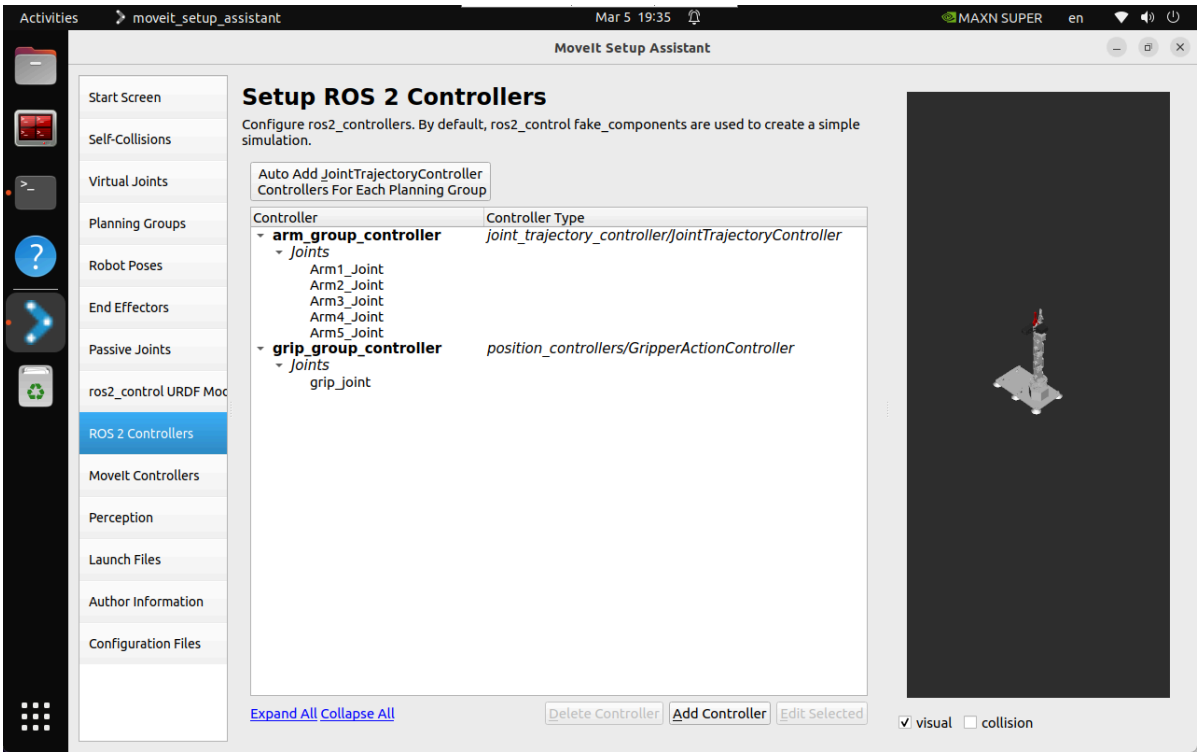


Add the gripper controller:

- Controller Name：grip_group_controller
- Controller Type：position_controllers/GripperActionController
- Add Planning Group Joints：grip_group


Activities
moveit_setup_assistant
Mar 5 19:34
MAXN SUPER
en
MoveIt Setup Assistant
Start Screen
Self-Collisions
Virtual Joints
Planning Groups
Robot Poses
End Effectors
Passive Joints
ros2_control URDF Mod
ROS 2 Controllers
MoveIt Controllers
Perception
Launch Files
Author Information
Configuration Files
Setup ROS 2 Controllers
Configure ros2_controllers. By default, ros2_control fake_components are used to create a simple simulation.
Create New Controller
Controller Options
Controller Name: grip_group_controller
Controller Type: position_controllers/GripperActionController
1
Next, Add Components To Controller:
Recommended:
Add Planning Group Joints
2
Advanced Options:
Add Individual Joints
Save
Cancel
visual
collision

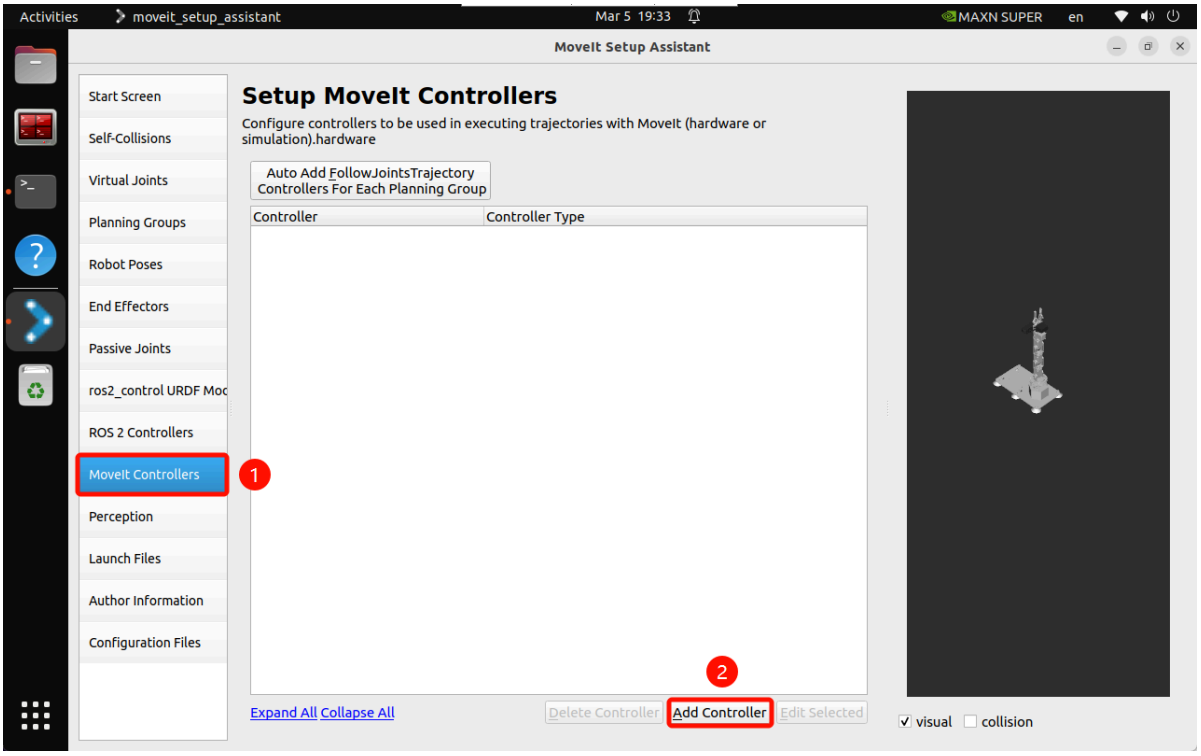

Activities
moveit_setup_assistant
Mar 5 19:35
MAXN SUPER
en
MoveIt Setup Assistant
Start Screen
Self-Collisions
Virtual Joints
Planning Groups
Robot Poses
End Effectors
Passive Joints
ros2_control URDF Mod
ROS 2 Controllers
MoveIt Controllers
Perception
Launch Files
Author Information
Configuration Files
Setup ROS 2 Controllers
Configure ros2_controllers. By default, ros2_control fake_components are used to create a simple simulation.
Edit 'grip_group_controller' Joints groups collection
Available Groups
Group Names
1 arm_group
2 grip_group
1
>
2
<
Selected Groups
Group Names
1 grip_group
Save
Cancel
3
visual
collision



2.10, MoveIt Controllers

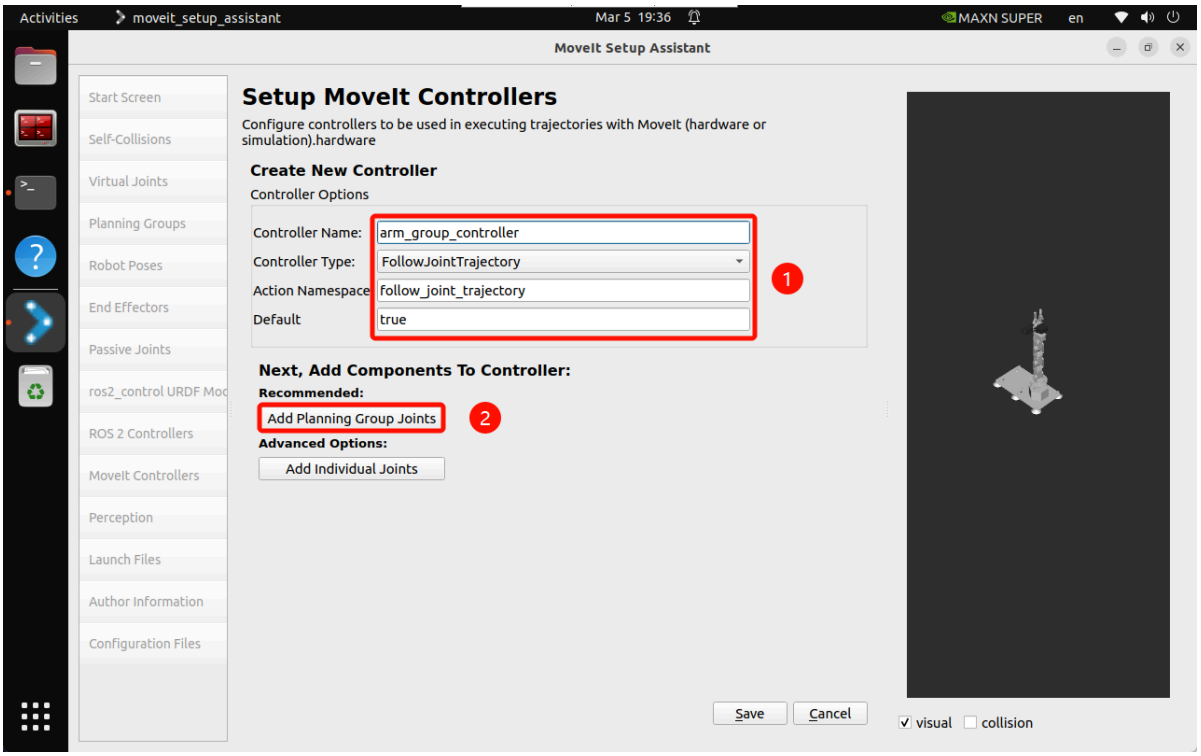
MoveIt requires a trajectory controller with the FollowJointTrajectoryAction interface to execute the planned trajectory, which sends the generated trajectory to the robot ROS2 controller.

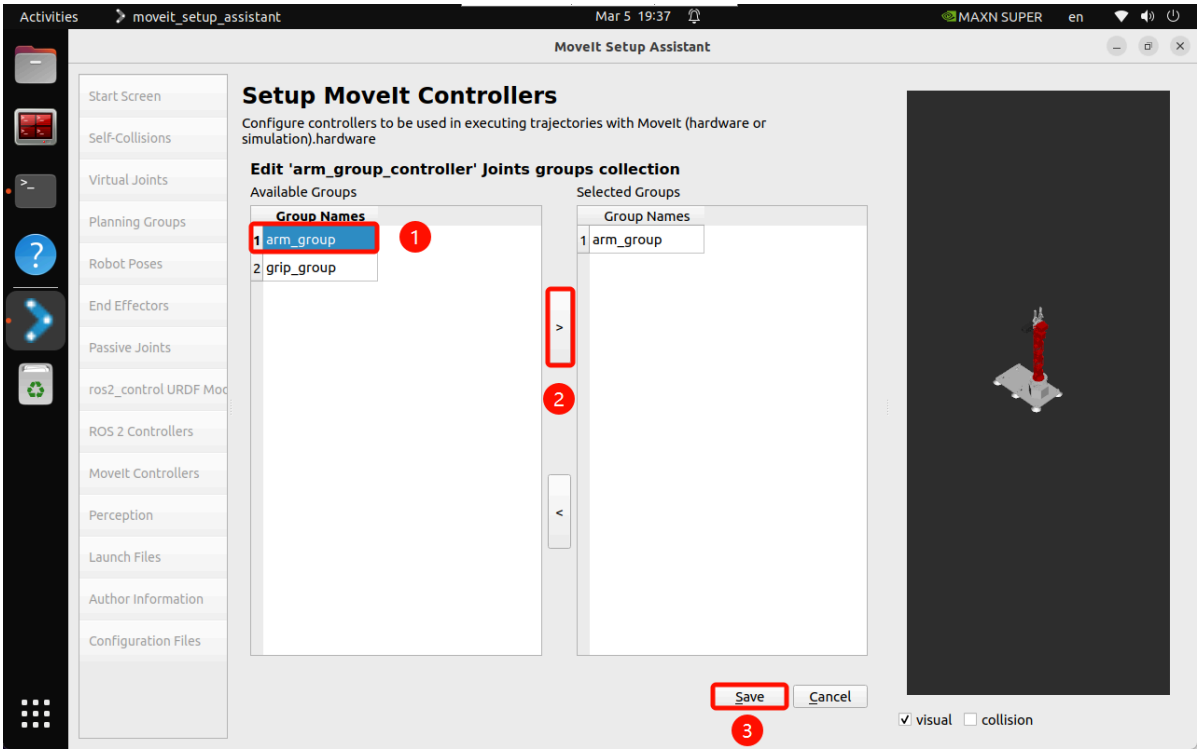
Added MoveIt Controllers need to ensure that the controller name matches the name configured in ROS2 Controllers.

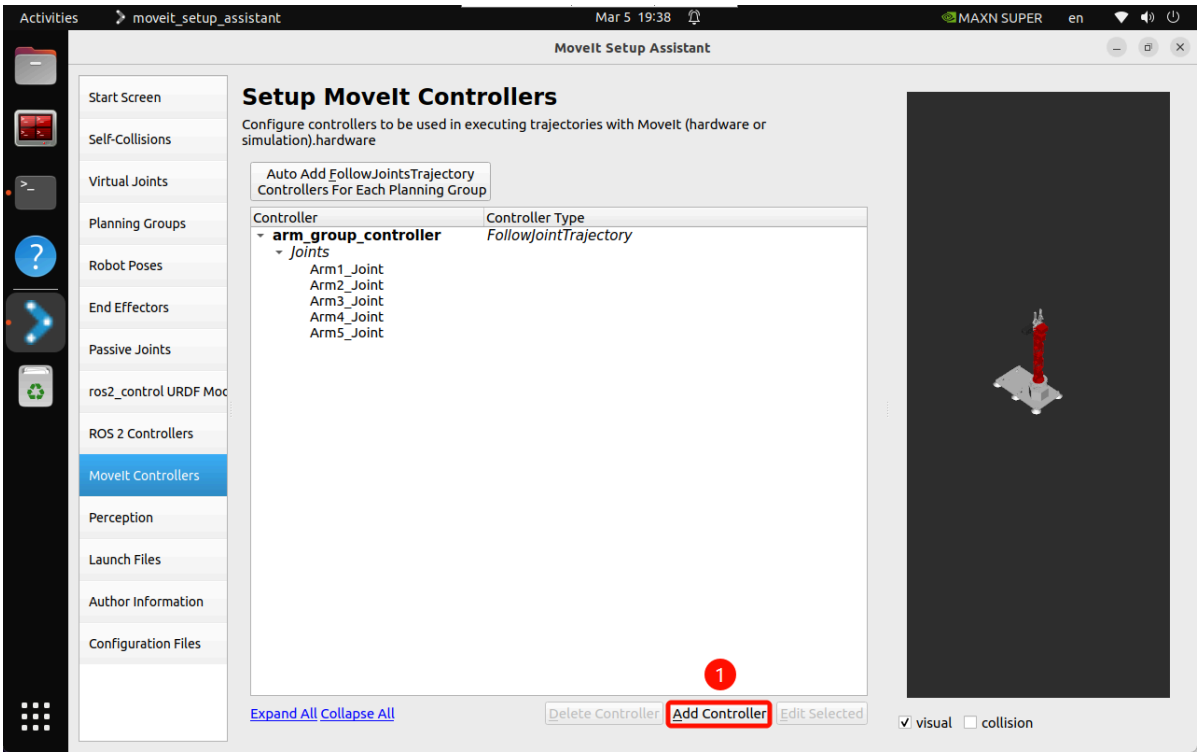


Add a robotic arm controller:

- Controller Name：arm_group_controller
- Controller Type：FollowJointTrajectory
- Add Planning Group Joints：arm_group

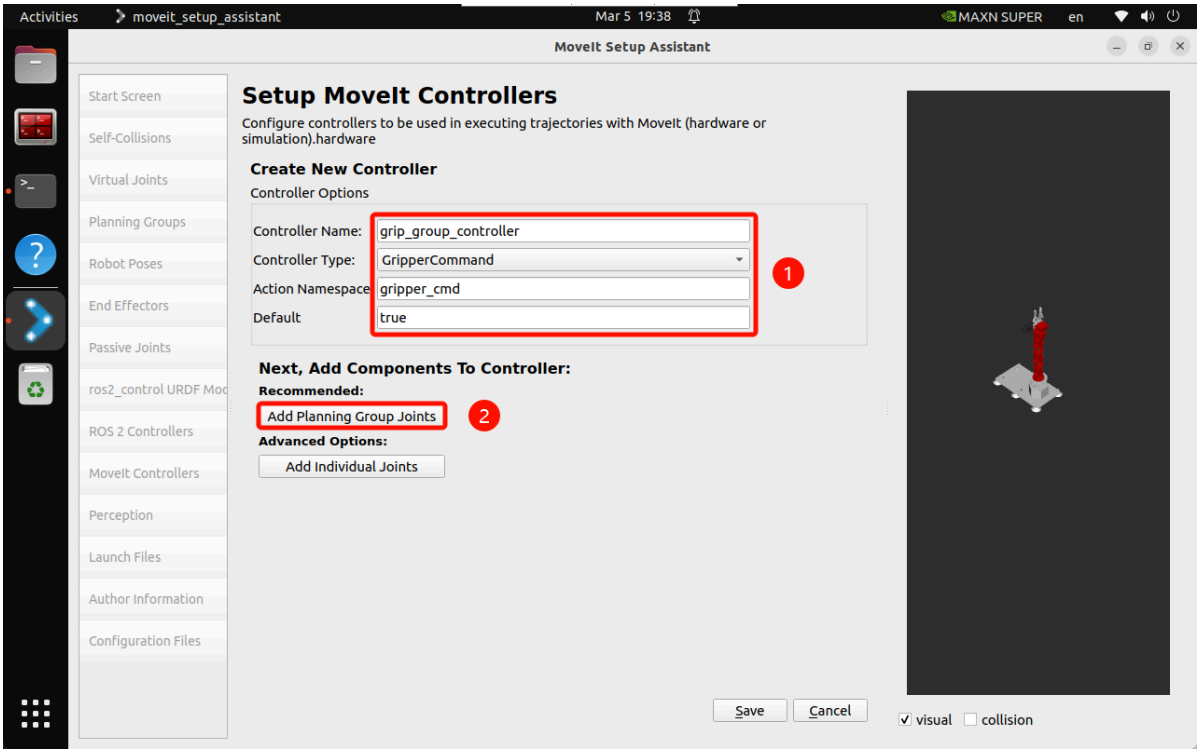

Activities
moveit_setup_assistant
Mar 5 19:36
MAXN SUPER
en
MoveIt Setup Assistant
Start Screen
Self-Collisions
Virtual Joints
Planning Groups
Robot Poses
End Effectors
Passive Joints
ros2_control URDF Moc
ROS 2 Controllers
MoveIt Controllers
Perception
Launch Files
Author Information
Configuration Files
Setup MoveIt Controllers
Configure controllers to be used in executing trajectories with MoveIt (hardware or simulation).hardware
Create New Controller
Controller Options
Controller Name: arm_group_controller
Controller Type: FollowJointTrajectory
Action Namespace: follow_joint_trajectory
Default: true
1
Next, Add Components To Controller:
Recommended:
Add Planning Group Joints
2
Advanced Options:
Add Individual Joints
Save
Cancel
visual
collision

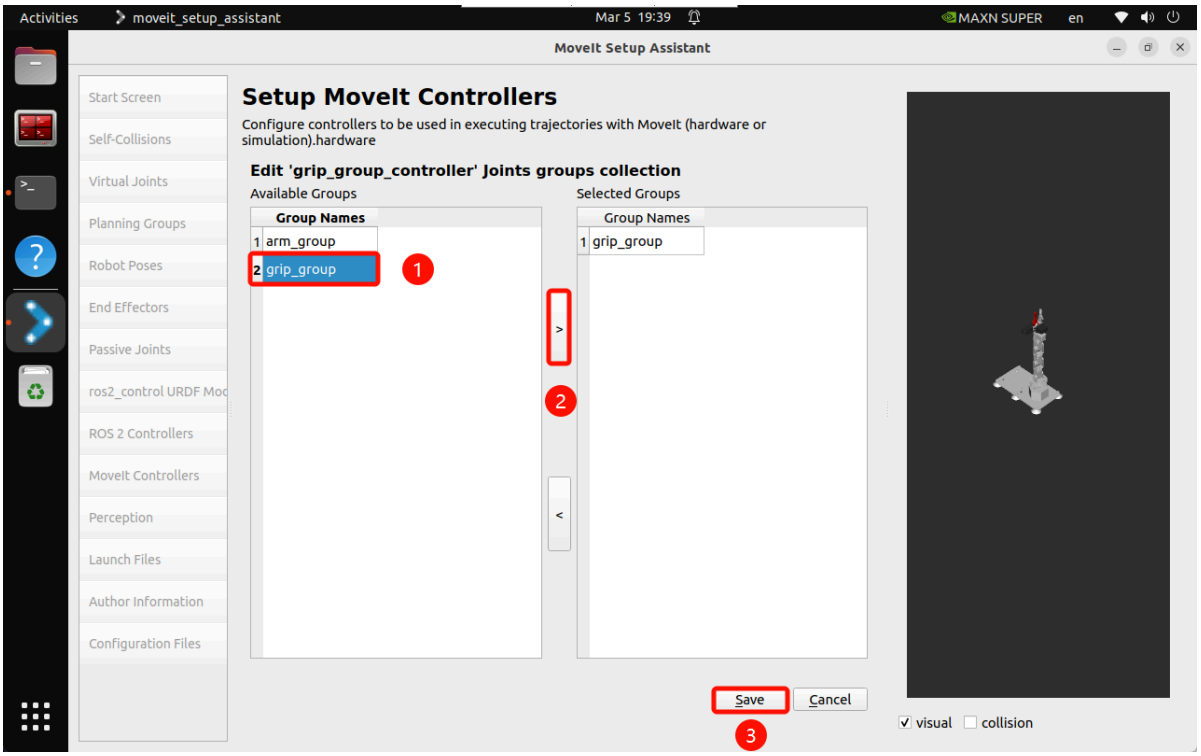

Activities
moveit_setup_assistant
Mar 5 19:37
MAXN SUPER
en
MoveIt Setup Assistant
Start Screen
Self-Collisions
Virtual Joints
Planning Groups
Robot Poses
End Effectors
Passive Joints
ros2_control URDF Moc
ROS 2 Controllers
MoveIt Controllers
Perception
Launch Files
Author Information
Configuration Files
Setup MoveIt Controllers
Configure controllers to be used in executing trajectories with MoveIt (hardware or simulation).hardware
Edit 'arm_group_controller' Joints groups collection
Available Groups
Group Names
1 arm_group
2 grip_group
1
>
2
<
Selected Groups
Group Names
1 arm_group
Save
Cancel
3
visual
collision

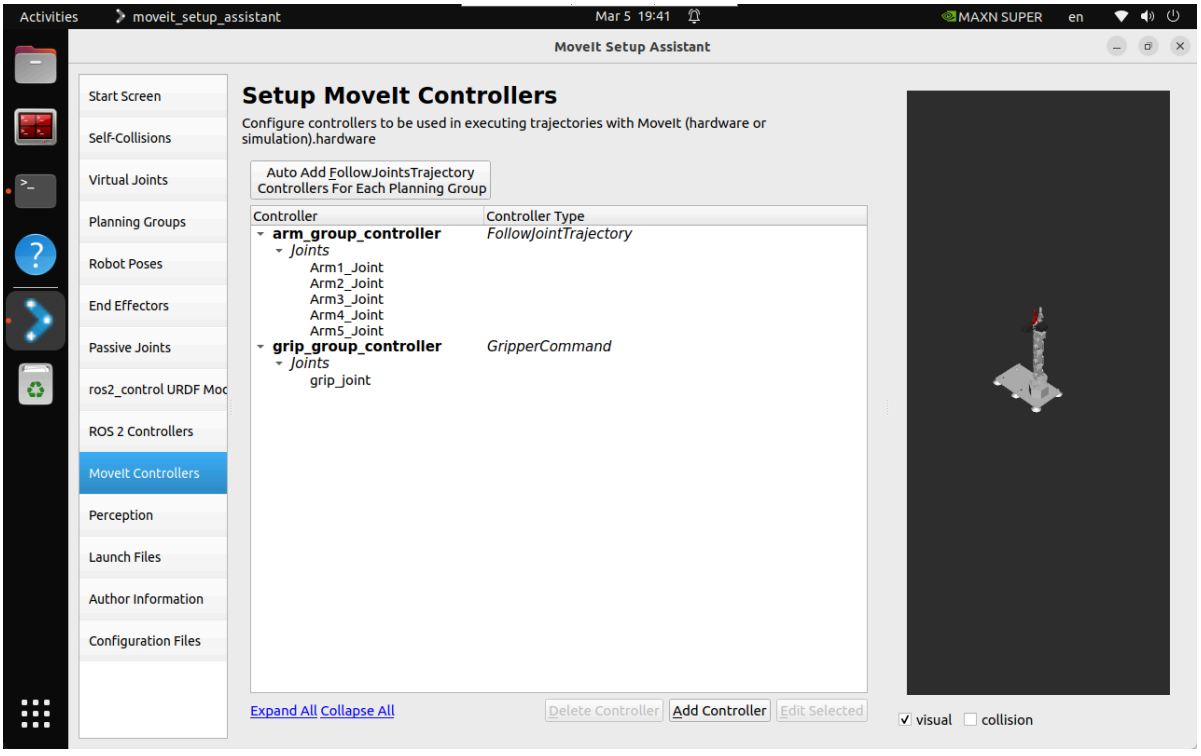


Add the gripper controller:

- Controller Name：grip_group_controller
- Controller Type：Gripper Command
- Add Planning Group Joints：grip_group

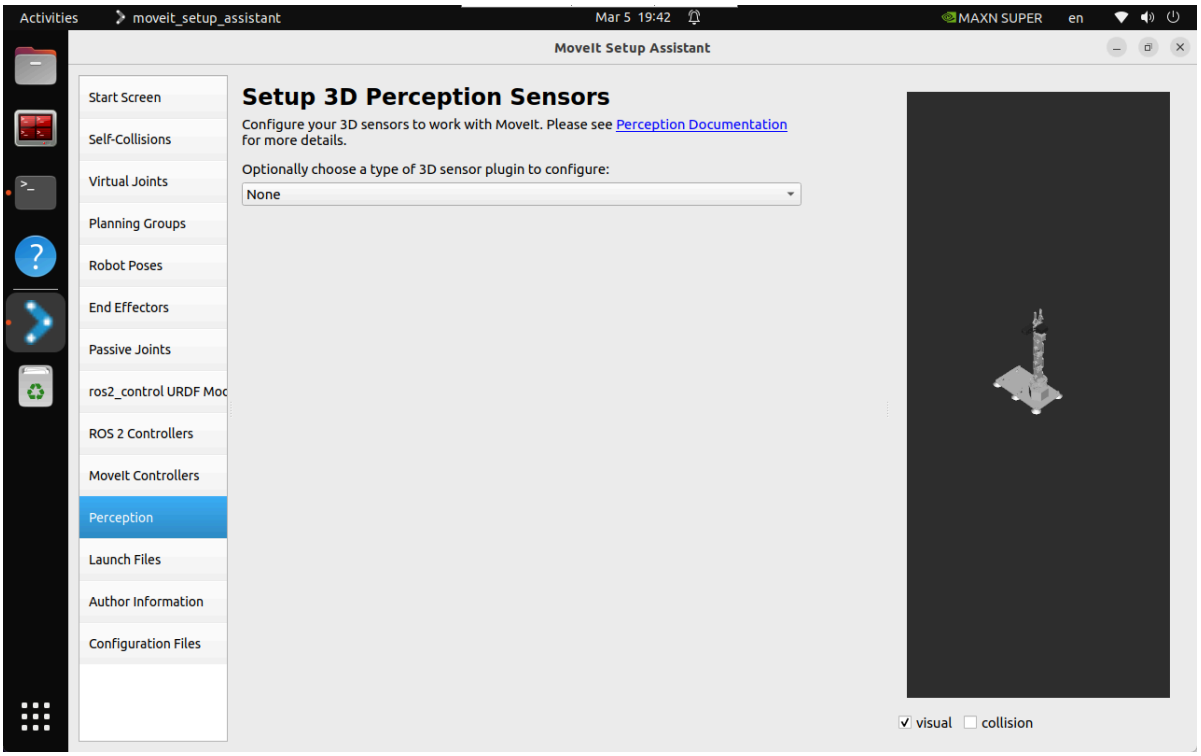






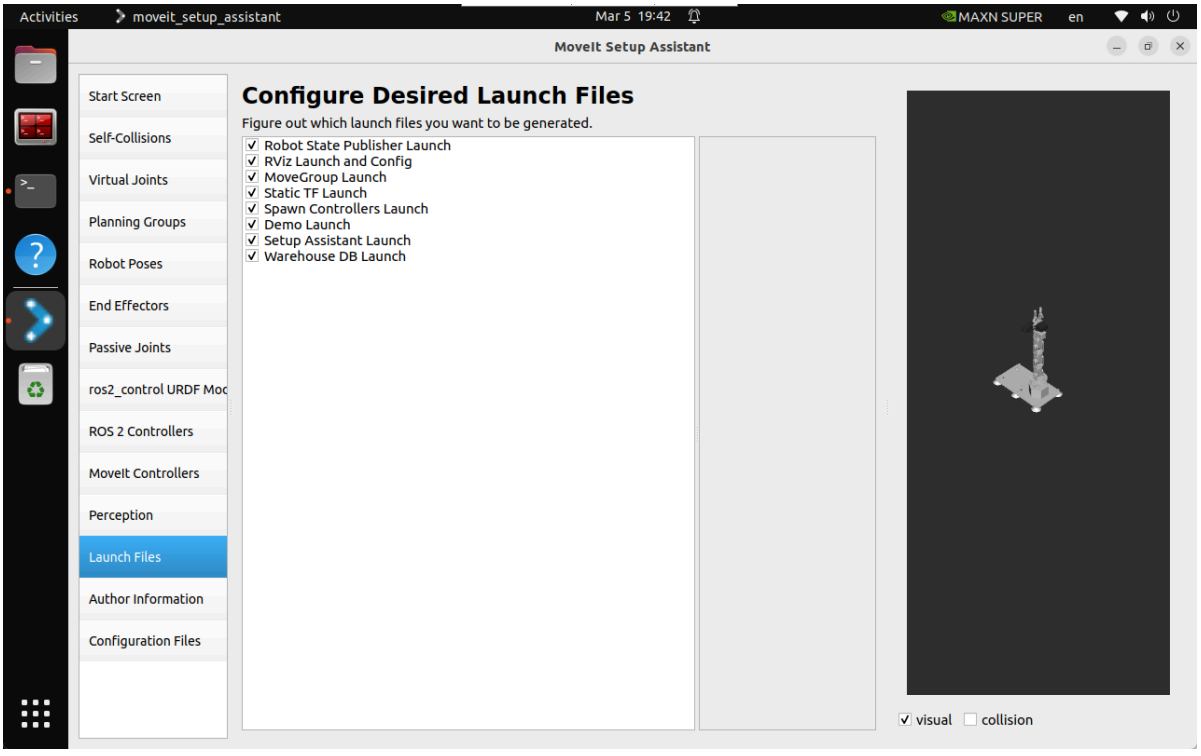
2.11. Sensors

Configure the settings of the 3D sensor used by the robot: No configuration is required.



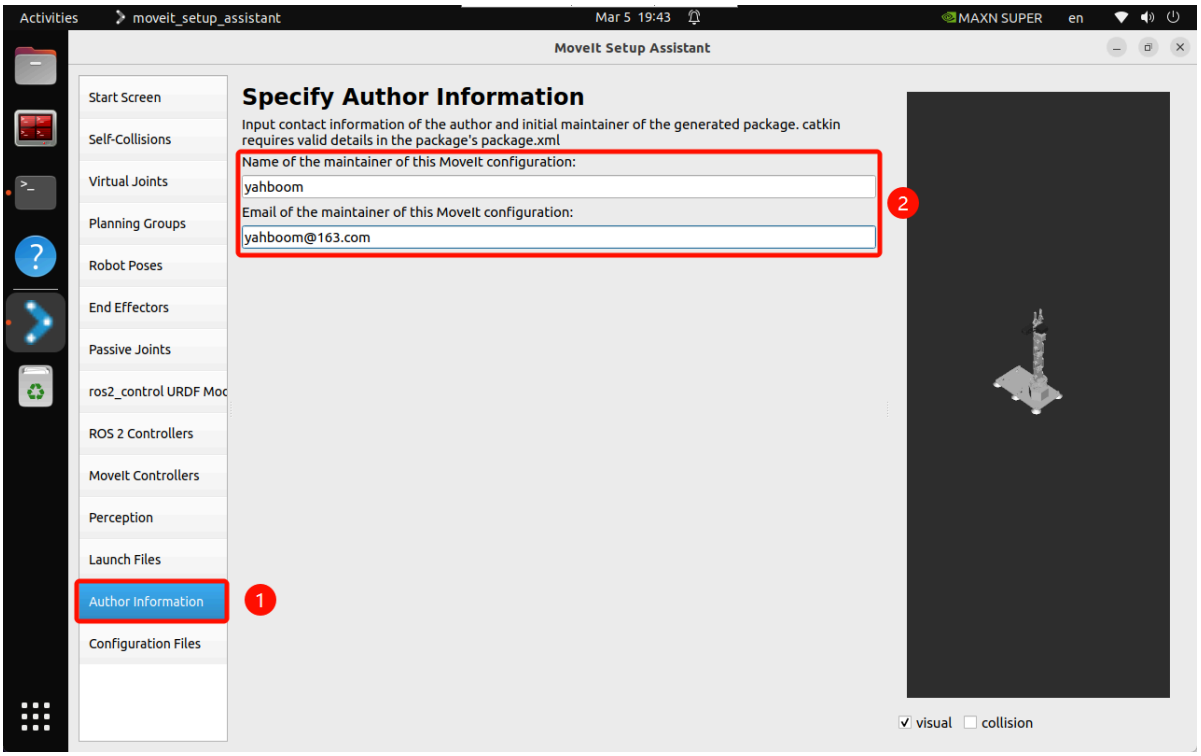
2.12, Startup file

Configure the automatically generated startup file: Use the default options.



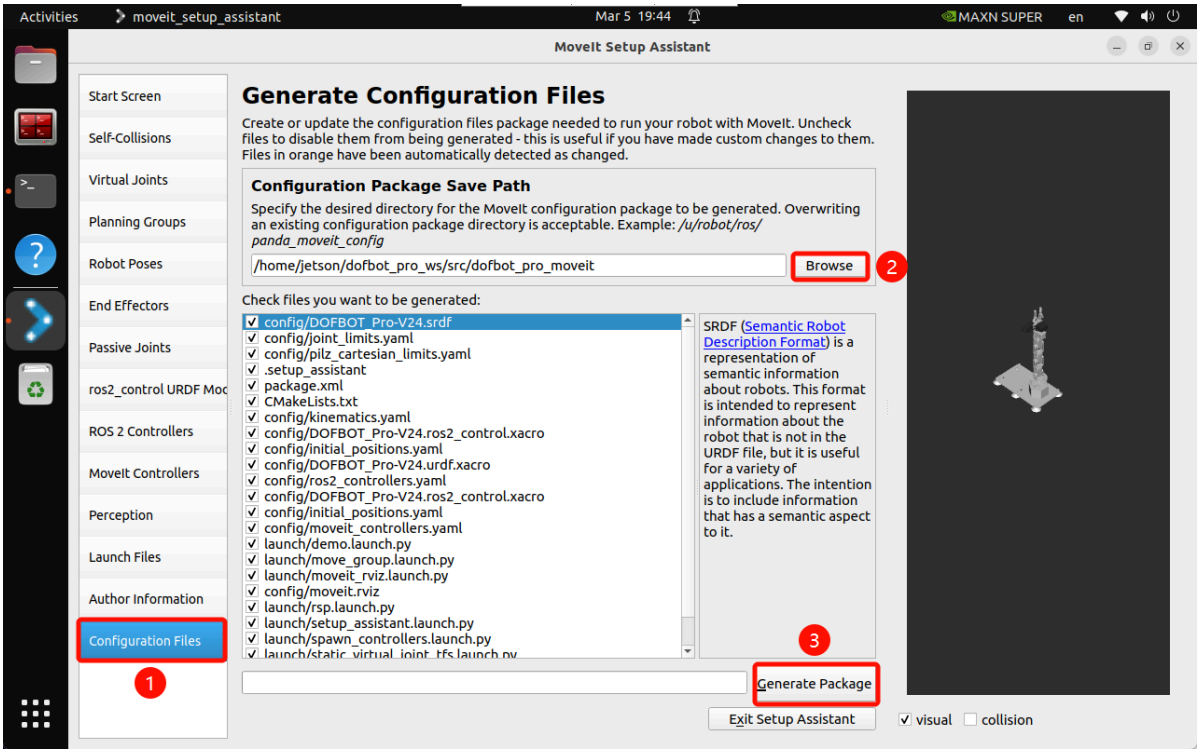
2.13, Author information

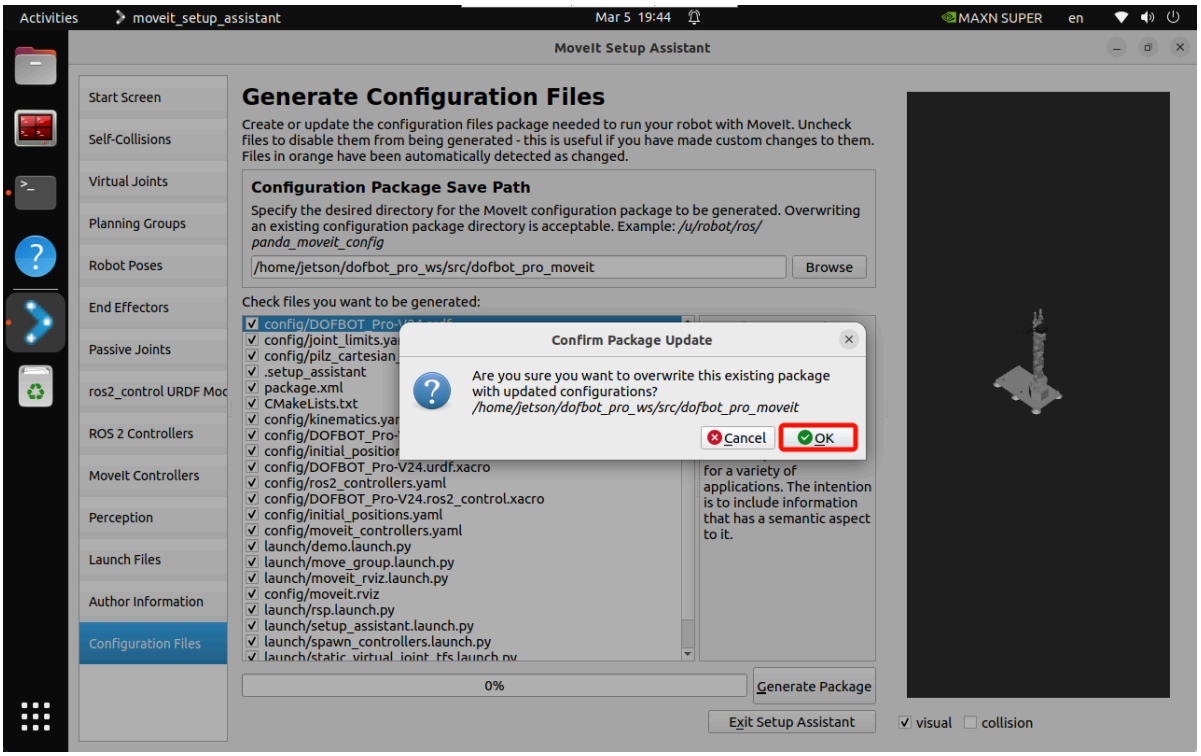
Add author information for software package generation:

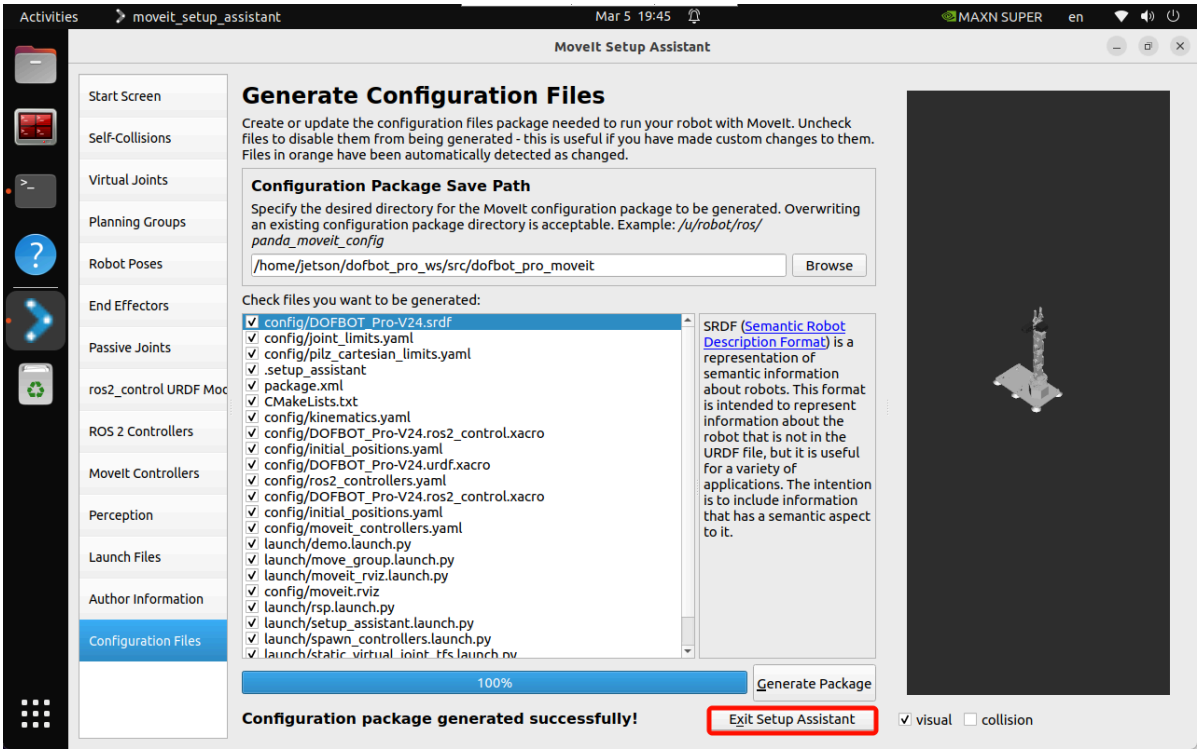


2.14. Generate configuration

Generate configuration files to the specified folder:

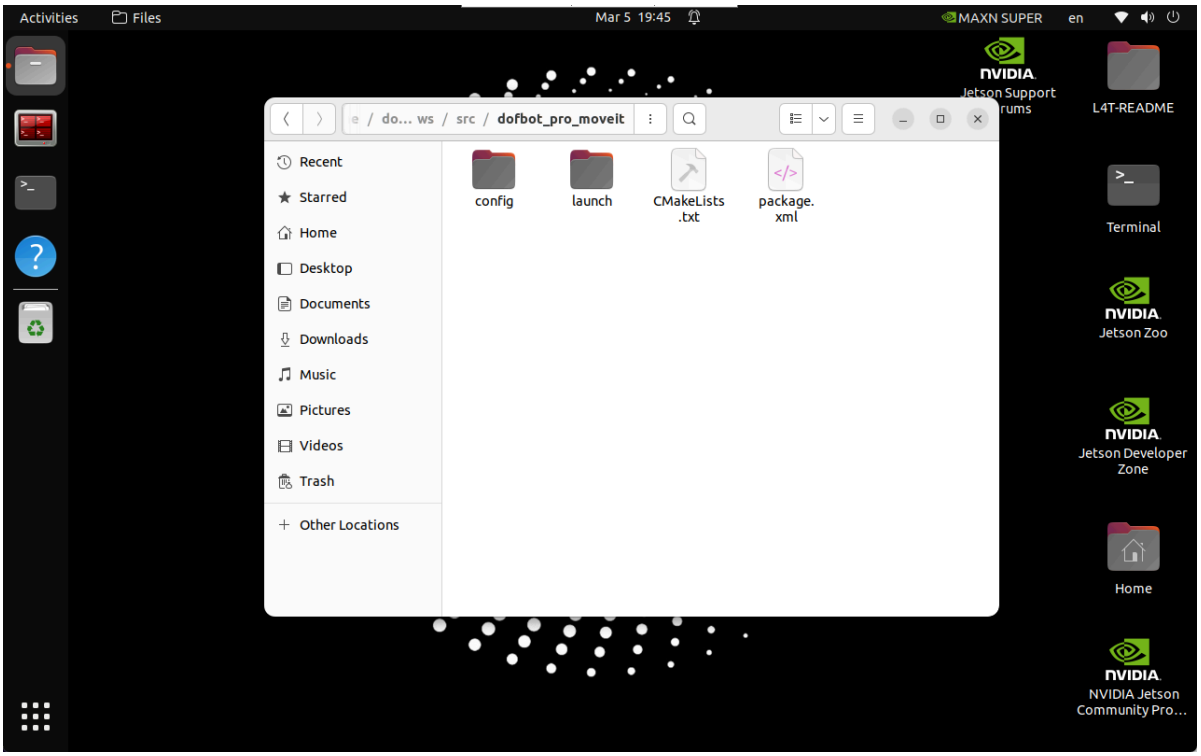






3. Configuration file

Enter the `dofbot_pro_moveit` folder:



4. Configuration Verification

The configuration file generated by MoveIt Setup Assistant needs to be modified to remove the warnings and model loading issues during startup.

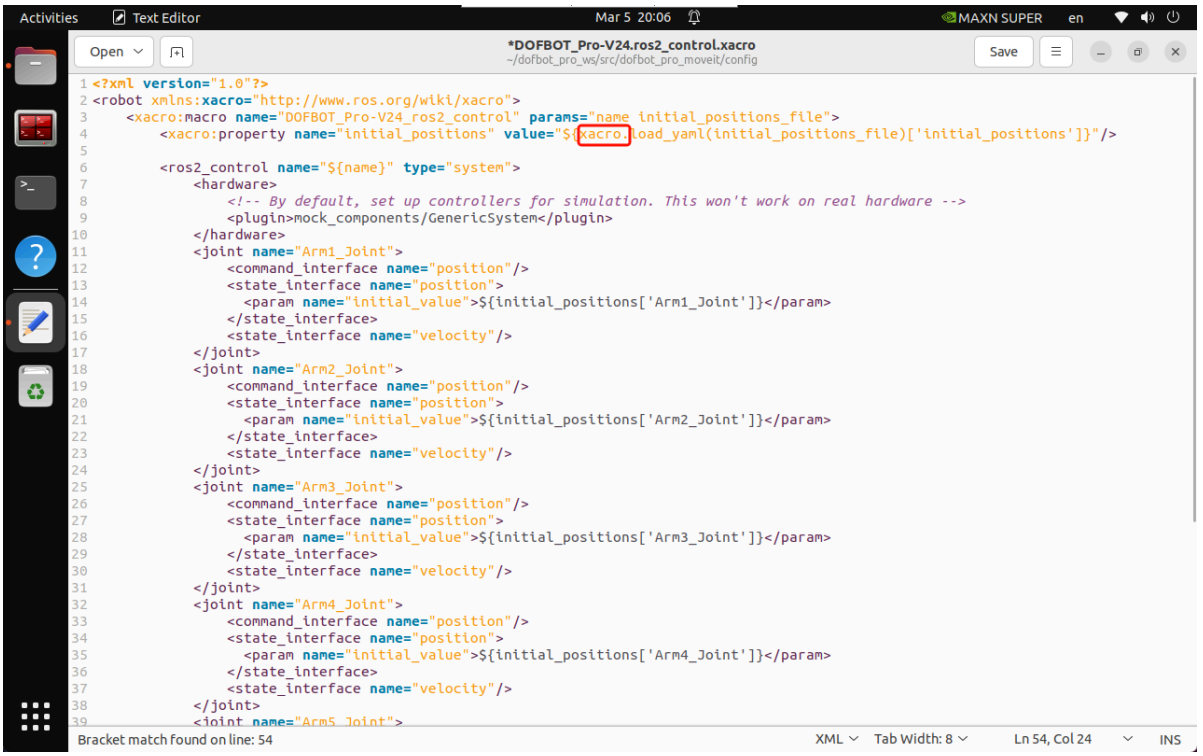
4.1. Modify the configuration file

> **DOFBOT_Pro-V24.ros2_control.xacro**

Modify the `DOFBOT_Pro-V24.ros2_control.xacro` file: modify the function of loading yaml

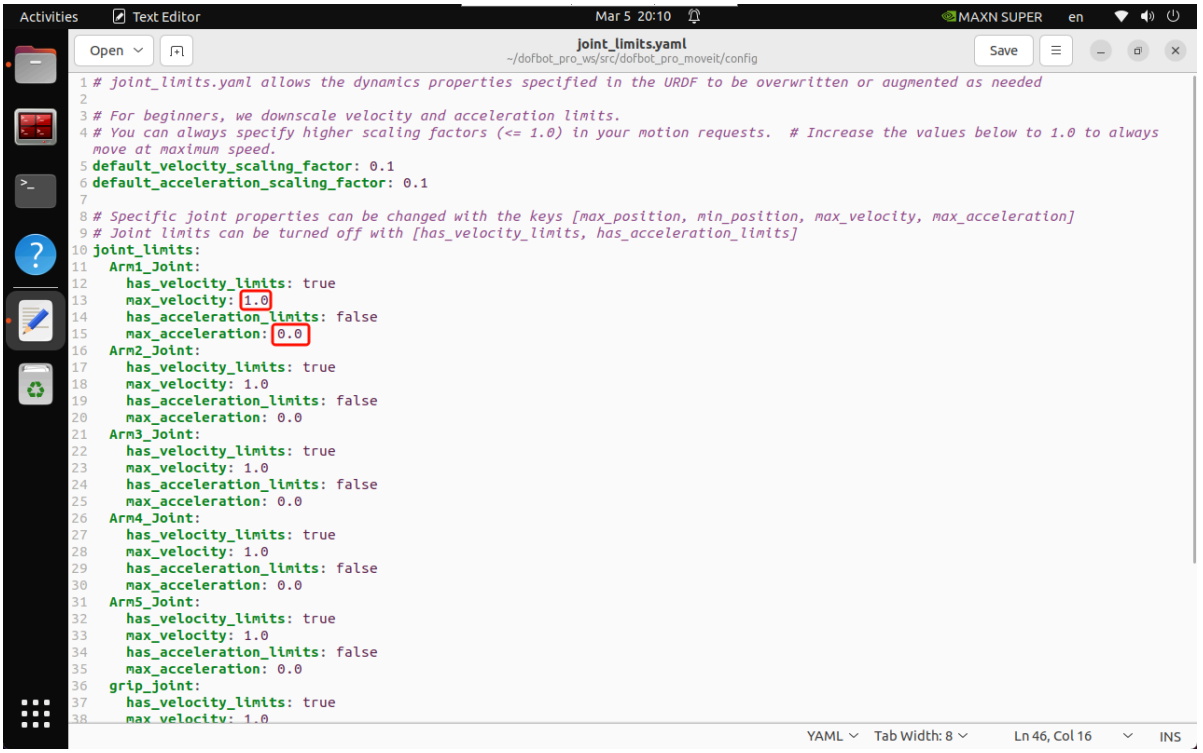
```
load_yaml(initial_positions_file)['initial_positions']

xacro.load_yaml(initial_positions_file)['initial_positions']
```



> **joint_limits.yaml**

Modify the `joint_limits.yaml` file: change all joint maximum velocities and accelerations to decimals



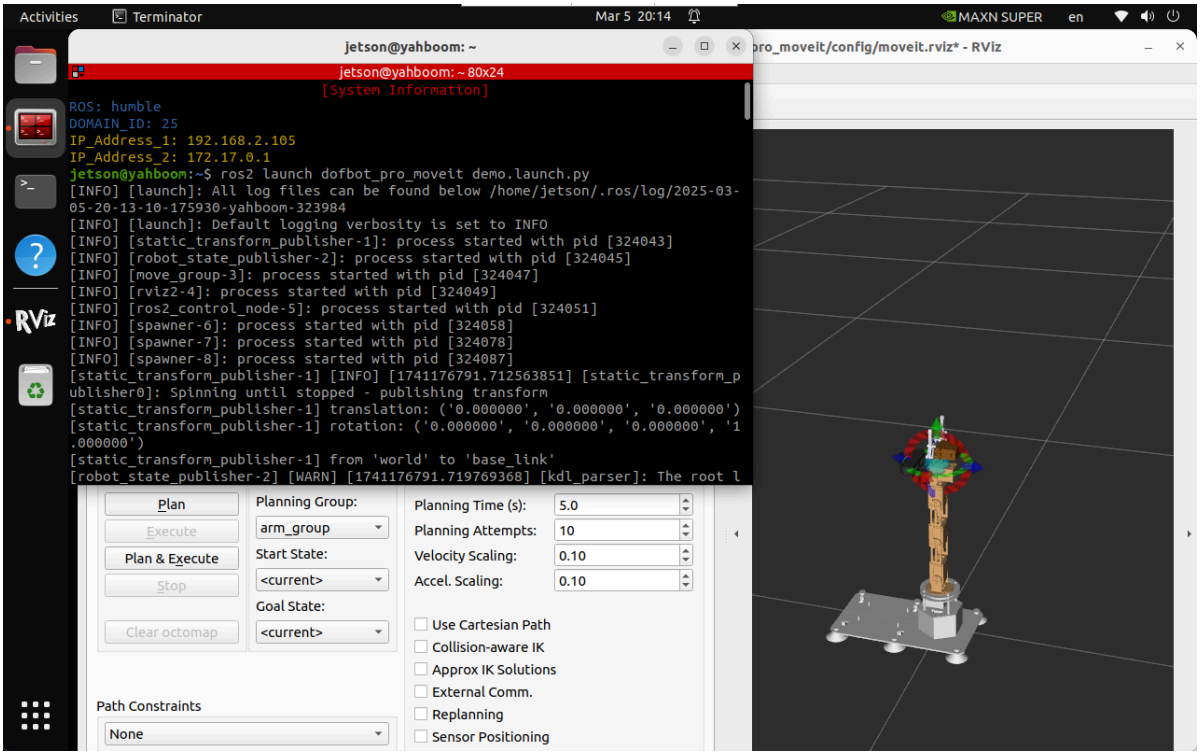
4.2. Compile the software package

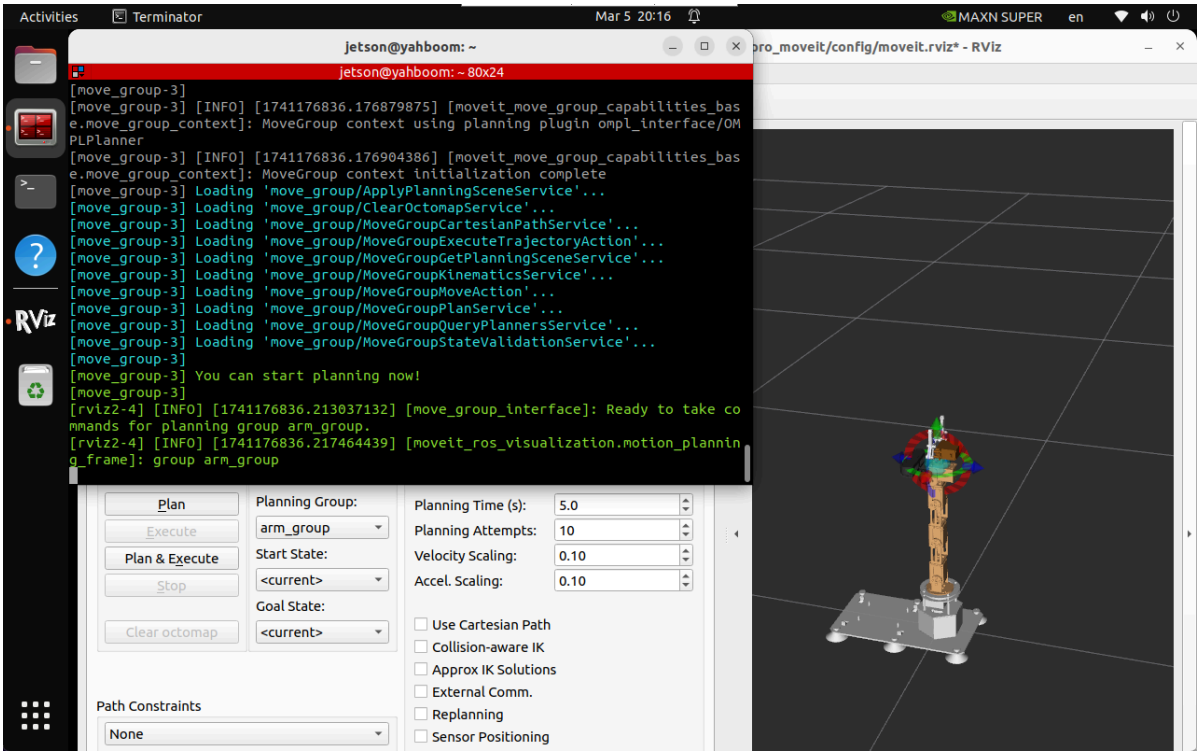
```
cd ~/dofbot_pro_ws
colcon build --packages-select dofbot_pro_moveit
source install/setup.bash
```

4.3. Start MoveIt

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

Starting the simulation is slow. Wait until the terminal displays `You can start planning now!` or the robot arm has a trackball (the trackball is a new sphere on the robot arm), indicating that the loading is complete.





References

MoveIt2 Humble：https://moveit.picknik.ai/humble/index.html

MoveIt2：https://moveit.picknik.ai/main/index.html