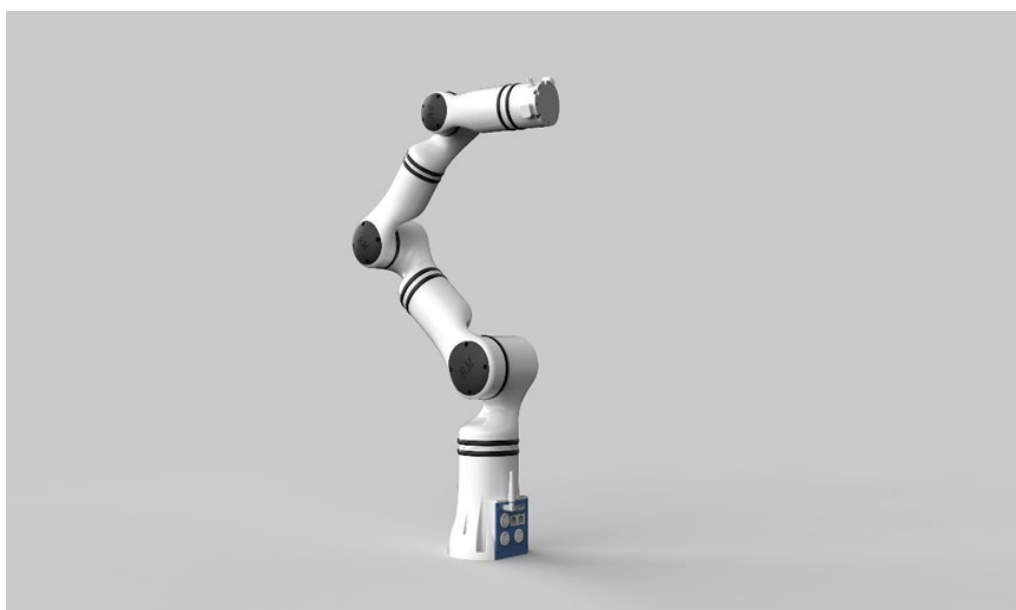




睿尔曼机器人 rm_gazebo 使用说明书 V1.0



睿尔曼智能科技（北京）有限公司



文件修订记录：

版本号	时间	备注
V1.0	2024-1-17	拟制



目录

1. rm_gazebo 功能包说明	3
2. rm_gazebo 功能包运行	3
2.1 控制仿真机械臂	3
3. rm_gazebo 功能包架构说明	4
3.1 功能包文件总览	4



1. rm_gazebo 功能包说明

rm_gazebo 的主要作用为帮助我们实现机械臂 Moveit 规划的仿真功能，我们将在 gazebo 的仿真环境中搭建一个虚拟机械臂，然后通过 Moveit 控制 gazebo 中的虚拟机械臂，在下文中将通过以下几个方面详细介绍该功能包。

1. 功能包使用。
2. 功能包架构说明。

通过这部分内容的介绍可以帮助大家：

1. 了解该功能包的使用。
2. 熟悉功能包中的文件构成及作用。

源码地址：https://github.com/RealManRobot/rm_robot/tree/main/rm_gazebo。

2. rm_gazebo 功能包运行

2.1 控制仿真机械臂

在完成环境安装和功能包安装后，我们可以进行 rm_gazebo 功能包的运行。

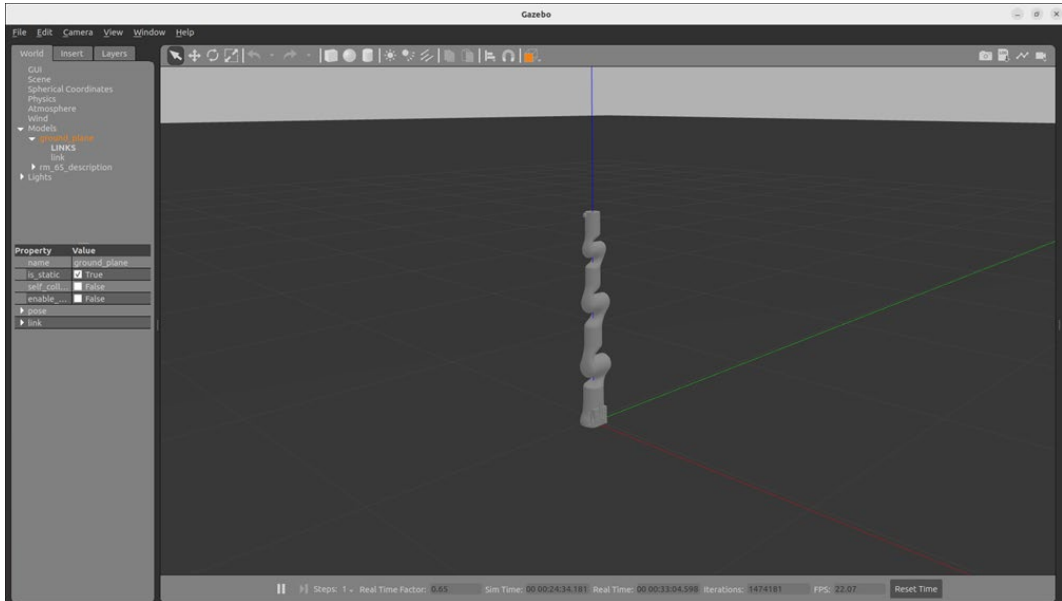
在运行之前首先我们需要修改有关配置文件，找到如下路径中的 rm_<arm_type>_moveit_controller_manager.launch.xml 文件，将下方的红框代码取消注释，注释之前的 yaml 加载代码。

```
1 <launch>
2   <arg name="fake_execution_type" default="FollowJointTrajectory" />
3   <!-- loads moveit controller manager on the parameter server which is taken as argument
4   if no argument is passed, moveit_simple_controller_manager will be set -->
5   <arg name="moveit_controller_manager"
6   default="moveit_simple_controller_manager/MoveItSimpleControllerManager" />
7   <param name="moveit_controller_manager" value="$(arg moveit_controller_manager)"/>
8   <!-- load controller list -->
9   <arg name="use_controller_manager" default="true" />
10  <param name="use_controller_manager" value="$(arg use_controller_manager)" />
11  <!-- loads ros controllers to the param server -->
12  <!-- <rosparam file="$(find rm_65_moveit_config)/config/ros_controllers.yaml"/> -->
13  <!-- <rosparam file="$(find rm_65_moveit_config)/config/controllers.yaml"/> -->
14  <!-- <rosparam file="$(find rm_65_moveit_config)/config/controllers_gazebo.yaml"/> -->
15  </launch>
```

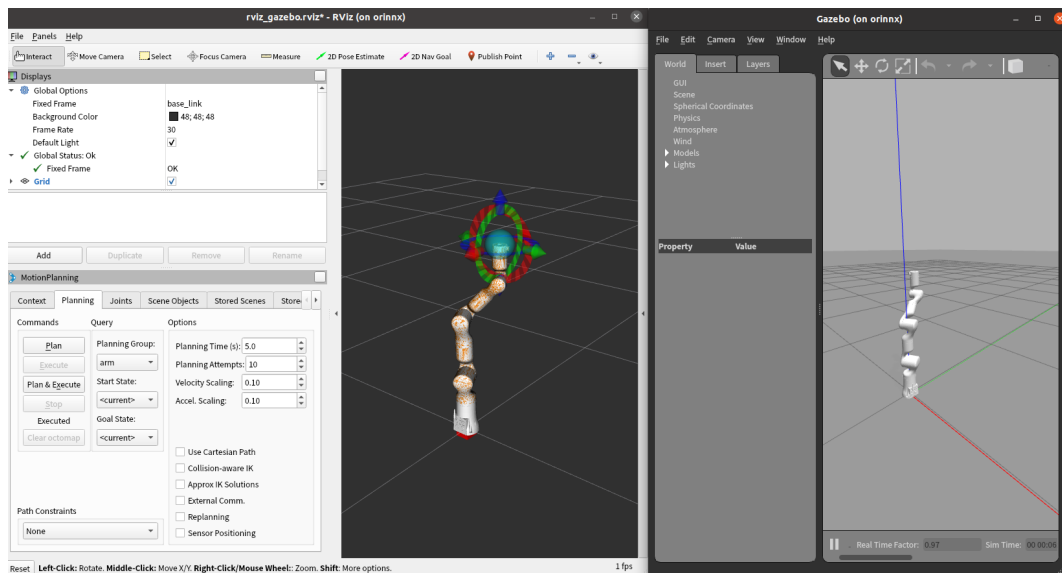
完成上面操作后，使用如下指令启动 gazebo 虚拟空间和虚拟机械臂。

```
rm@rm-desktop:~$ roslaunch rm_gazebo arm_<arm_type>_bringup_moveit.launch
```

运行成功后将弹出如下界面。



弹出 rviz 的控制界面后就可以进行 Moveit 和 gazebo 的仿真控制了。



3. rm_gazebo 功能包架构说明

3.1 功能包文件总览

当前 rm_gazebo 功能包的文件构成如下。

└─ CMakeLists.txt	#编译规则文件
└─ config	
├─ ECO65	#ECO65 的仿真配置文件
├─ arm_gazebo_control.yaml	



```
| | └─ arm_gazebo_joint_states.yam    #关节状态控制器
| | └─ rm_eco65_trajectory_control.yaml #关节控制器
| | └─ trajectory_control.yaml
| └─ RM65                               #同 ECO65 的仿真配置文件
| | └─ arm_gazebo_control.yaml
| | └─ arm_gazebo_joint_states.yaml
| | └─ rm_65_trajectory_control.yaml
| | └─ rviz_gazebo.rviz
| | └─ trajectory_control.yaml
| └─ RM75                               #同 ECO65 的仿真配置文件
| | └─ arm_gazebo_control.yaml
| | └─ arm_gazebo_joint_states.yaml
| | └─ rm_75_trajectory_control.yaml
| | └─ rviz_gazebo_bak.rviz
| | └─ rviz_gazebo.rviz
| | └─ trajectory_control.yaml
| └─ RML63                               #同 ECO65 的仿真配置文件
|   └─ arm_gazebo_control.yaml
|   └─ arm_gazebo_joint_states.yaml
|   └─ rml_63_trajectory_control.yaml
|   └─ rviz_gazebo.rviz
|   └─ trajectory_control.yaml
```



```
└─ launch
  │ └─ arm_63_bringup_moveit.launch          #RML63 仿真 gazebo 启动文件
  │ └─ arm_65_bringup_moveit.launch          # RM65 仿真 gazebo 启动文件
  │ └─ arm_75_bringup_moveit.launch          # RM75 仿真 gazebo 启动文件
  │ └─ arm_eco65_bringup_moveit.launch       # ECO65 仿真 gazebo 启动文件
  │ └─ ECO65                                # ECO65 相关启动文件
  │ │ └─ arm_eco65_trajectory_controller.launch #仿真控制器 launch 文件
  │ │ └─ rm_eco65_arm_gazebo_states.launch    #仿真机械臂状态监控 launch 文件
  │ │ └─ rm_eco65_arm_trajectory_controller.launch
  │ │ └─ rm_eco65_arm_world.launch           #仿真机械臂 gazebo 加载 launch 文件
  │ └─ RM65                                # RM65gazebo 相关启动文件(同 eco65)
  │ │ └─ arm_65_trajectory_controller.launch
  │ │ └─ rm_65_arm_gazebo_states.launch
  │ │ └─ rm_65_arm_trajectory_controller.launch
  │ │ └─ rm_65_arm_world.launch
  │ └─ RM75                                # RM75gazebo 相关启动文件(同 eco65)
  │ │ └─ arm_75_trajectory_controller.launch
  │ │ └─ arm_gazebo_states.launch
  │ │ └─ arm_trajectory_controller.launch
  │ │ └─ arm_world.launch
  │ └─ RML63                                # RML63gazebo 相关启动文件(同 eco65)
  │ └─ arm_gazebo_states.launch
```



```
|   |— arm_trajectory_controller.launch
|   |— arm_world.launch
|   └─ rml_63_trajectory_controller.launch
└─ package.xml
```