



# 遨博 i5 机械臂使用说明

北京中科原动力科技有限公司

2021.12.23

文档历史

版本号	日期	内容	责任人
1.0	2021.12.23	初始版本，基于 ROS 的机械臂调试方法说明	杨顺

# Table of Contents

<b>1 简介 .....</b>	<b>6</b>
<b>2 环境配置 .....</b>	<b>6</b>
2.1 VMware 虚拟机方式 .....	6
2.2 在 Ubuntu18.04 配置 ROS 环境 .....	6
<b>3 ROS 下的机械臂控制简要介绍 .....</b>	<b>8</b>
<b>4 参考资料 .....</b>	<b>9</b>

# 1 简介

遨博机械臂支持如 c/c++/python/c# 等多种 SDK，在后期实际项目开发过程中，重要以 SDK 开发为主。本文档介绍基于遨博虚拟机机械臂操作和基于 ROS 的机械臂调试和轨迹规划节点的环境配置、基本操作和使用方法。

## 2 环境配置

遨博机械臂提供了 VMWare 虚拟机可供快速配置示教界面用于熟悉虚拟机操作，除部分与硬件相关功能无法使用外，整体操作方法和示教器实物完全一致；我们也可以在成熟的 Ubuntu 开发环境中安装 ROS 及其依赖包进行环境配置，通过 moveit 和 gazebo 对机械臂进行运动轨迹的规划和展示。

### 2.1 VMware 虚拟机方式

1. 在下载遨博虚拟机，网盘地址：<https://pan.baidu.com/s/14i72n66UeIIBIqD-X8720Q>，提取码：4u2n。下载完成后，在 win10 系统下解压即可；
2. 安装 VMware Workstation 后，打开虚拟机载入解压后的遨博虚拟机，虚拟机内存、硬盘等配置方法可自行搜索；
3. 开启虚拟机进入系统界面，双击桌面上的 AUBORPE 即可进入与示教系统完全一致的编程环境；
4. 接受“机器人通讯错误进入仿真模式”的提示，进入示教仿真。

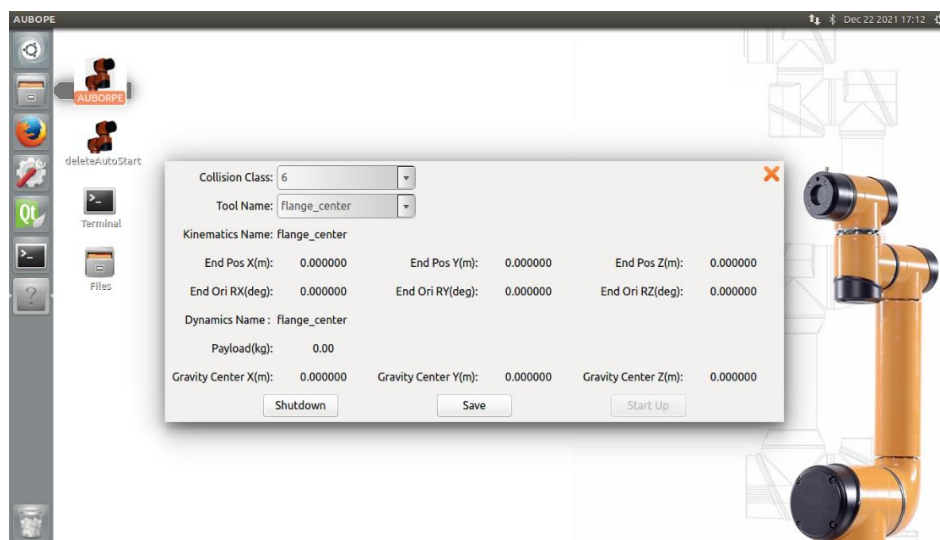


图 1 遨博虚拟机界面示意图

### 2.2 在 Ubuntu18.04 配置 ROS 环境

1. 安装 ROS，具体步骤参照 ROS 官网 [melodic/Installation/Ubuntu - ROS Wiki](http://wiki.ros.org/melodic/Installation/Ubuntu);

2. 安装 moveit-core 和 industrial: `sudo apt install ros-melodic-industrial-core;`
3. 如果要运行 gazebo, 安装以下依赖包:  
`sudo apt-get install ros-melodic-transmission-interface`  
`sudo apt-get install ros-melodic-gazebo-ros-control`  
`sudo apt-get install ros-melodic-joint-state-controller`  
`sudo apt-get install ros-melodic-effort-controllers`  
`sudo apt-get install ros-melodic-position-controllers`  
`sudo apt-get install ros-melodic-moveit-visual-tools`
4. 源码编译,  
`mkdir -p catkin_ws/src;`  
`cd catkin_ws/src;`  
`git clone -b melodic https://github.com/AuboRobot/aubo_robot;`  
在 src 文件夹中 `rosdep install --from-paths src --ignore-src -r -y;`  
`catkin_make`
5. 编译完成后, source devel/setup.bash, 然后启动 moveit:  
`roslaunch aubo_i5_moveit_config moveit_planning_execution.launch`  
`robot_ip:=127.0.0.1`
6. 如果想要在 gazebo 中观察机械臂运动状态, 则启动 gazebo:  
`roslaunch aubo_gazebo aubo_i5_gazebo_control.launch`  
若要在真实场景中使用 ROS-moveit 进行机械臂控制, 只需简单改变:  
`roslaunch aubo_i5_moveit_config moveit_planning_execution.launch`  
`sim:=false robot_ip:=192.168.0.123(real IP of robot)`

注意, 控制机械臂的过程中可以通过开关对控制权进行切换:

`rostopic pub -1 aubo_driver/controller_switch std_msgs/Int32 -0 (0 为非 ROS 控制, 1 为 ROS 控制)`, 另外, 在机械臂底座合理设计之前, 测试一定要保证规划的任意轴轨迹在底座高度以上!!!

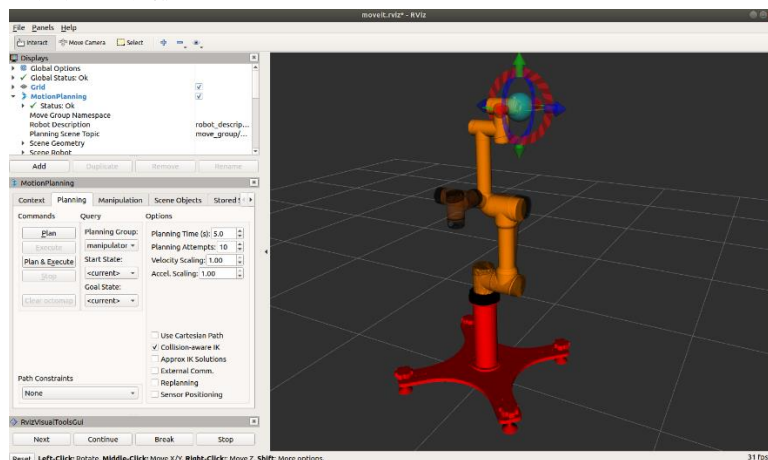


图 2 Rviz 下的 aubo i5 机械臂及 moveit 规划栏

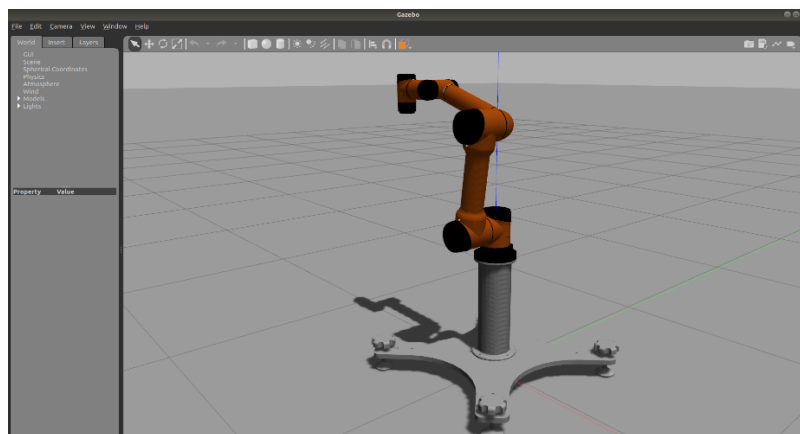


图 3 Gazebo 下的 aubo i5 机械臂

### 3 ROS 下的机械臂控制简要介绍

在 ROS 下进行机械臂控制，当前主要使用 Moveit，Moveit 是针对机械臂、部分机械手和移动机器人的开源平台，当前已经支持了超过 [60 种](#) 机器人的建模和轨迹控制，这些机械臂可以是真实的，也可以是虚拟仿真的。下面简要介绍基于 Moveit 的控制思路。

1. 首先要有一个与真实机械臂相一致的机器人 URDF 模型文件。这里一致指的是机械臂连杆参数一致，模型的细微差别不影响。这一步也不是必须的，只是为了在 rviz 下使用 moveit，这样可视化效果好一些，初学也容易接受，若不在 rviz 下使用 moveit，也完全可以使用 moveit 所提供的 API 编程接口，开发程序去发布规划的轨迹；
2. 使用 moveit 配置助手(Setup Assistant)生成配置文件。该步骤核心是定义虚拟节点、设定碰撞免检矩阵、**规划组**（重要!!）、定义关节、手臂、末端等；
3. 修改配置文件来控制真实机械臂。Moveit 配置文件夹提供了一些配置文件，但仅针对虚拟机械臂，如果要在真实世界中控制机械臂，需要相应做出修改；
4. 理解 moveit action 的工作原理，通过 action 交互机制来控制机械臂。

在实际使用过程中，由于遨博提供了官方的 ROS 包，里面已经针对 I 系列机械臂生成了配置文件，而且将控制封装成了很简单的函数，通过加载规划组、设定目标位姿或关节姿态，调用 `move_group.setPoseTarget()` / `move_group.plan()` / `move_group.execute(plan)` / `move_group.move()` 即可实现对机械臂的基本操控。同时，在规划控制过程中，可以设置路径约束、路径插值粒度、设定碰撞物体的位

姿等来实现更灵活的机械臂路径自主规划和控制。

如果需要更细致了解 `moveit` 下的控制方法，可以基于步骤 2 运行：

```
roslaunch aubo_demo MoveGroupInterface_To_Melodic.launch
```

观察 `rviz` 机械臂仿真运动与设定是否相同，也可查看其源代码熟悉 API 使用方法。

## 4 参考资料

[1] `moveit` 控制源代码：[aubo\\_robot/MoveGroupInterface\\_To\\_Melodic.cpp at melodic · AuboRobot/aubo\\_robot \(github.com\)](https://github.com/AuboRobot/aubo_robot/blob/master/melodic/MoveGroupInterface_To_Melodic.cpp)

[2] [使用 `moveit!` 控制真实机械臂（0）——思路 爱学习的草莓熊-CSDN 博客 `moveit` 控制真实机械臂](#)

[3] 遨博官方 ROS 仓库：[https://github.com/AuboRobot/aubo\\_robot](https://github.com/AuboRobot/aubo_robot)