

【ROS机械臂入门教程】

第7讲 Moveit基础(C++)

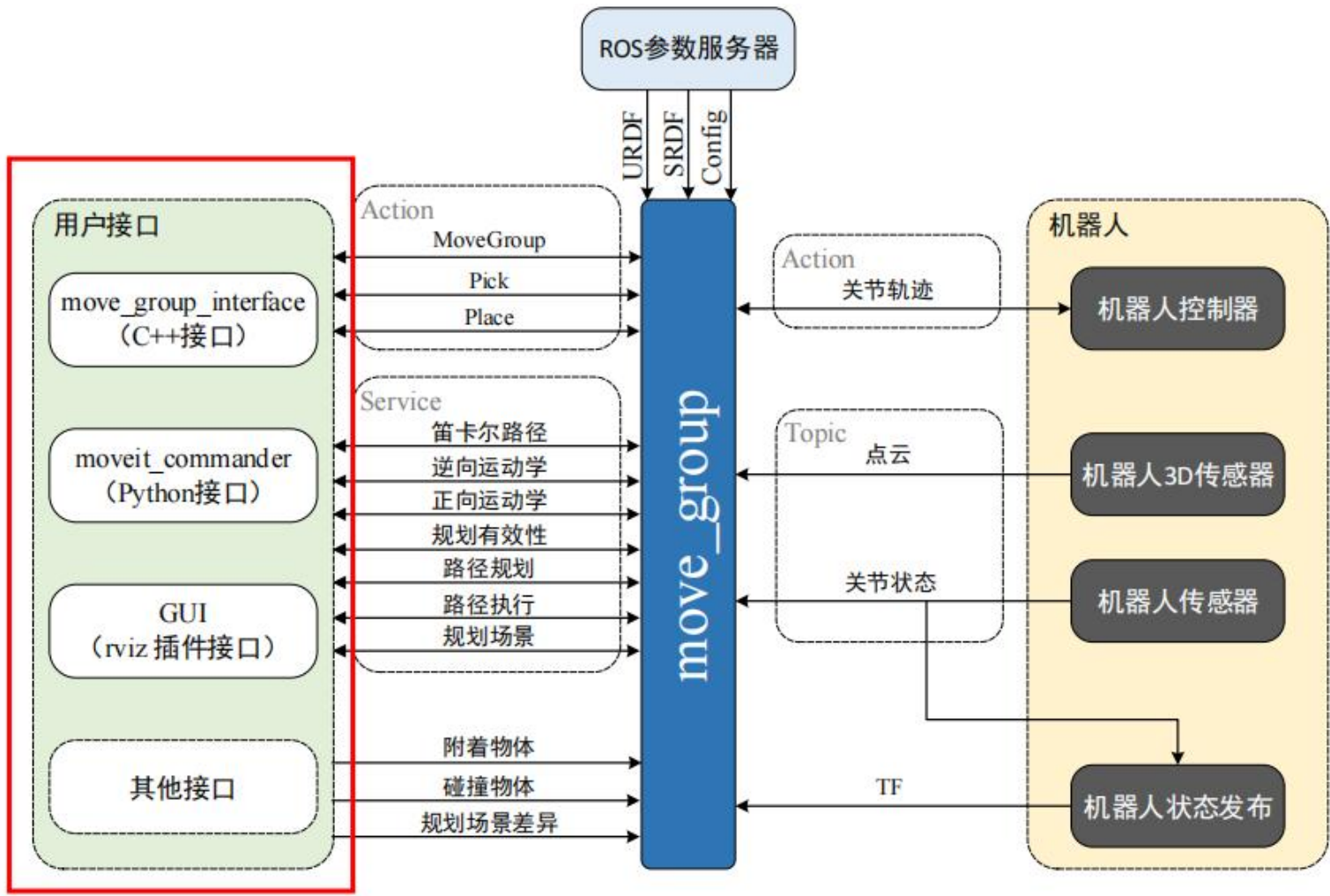
小五

日期 2023/1/28

目录

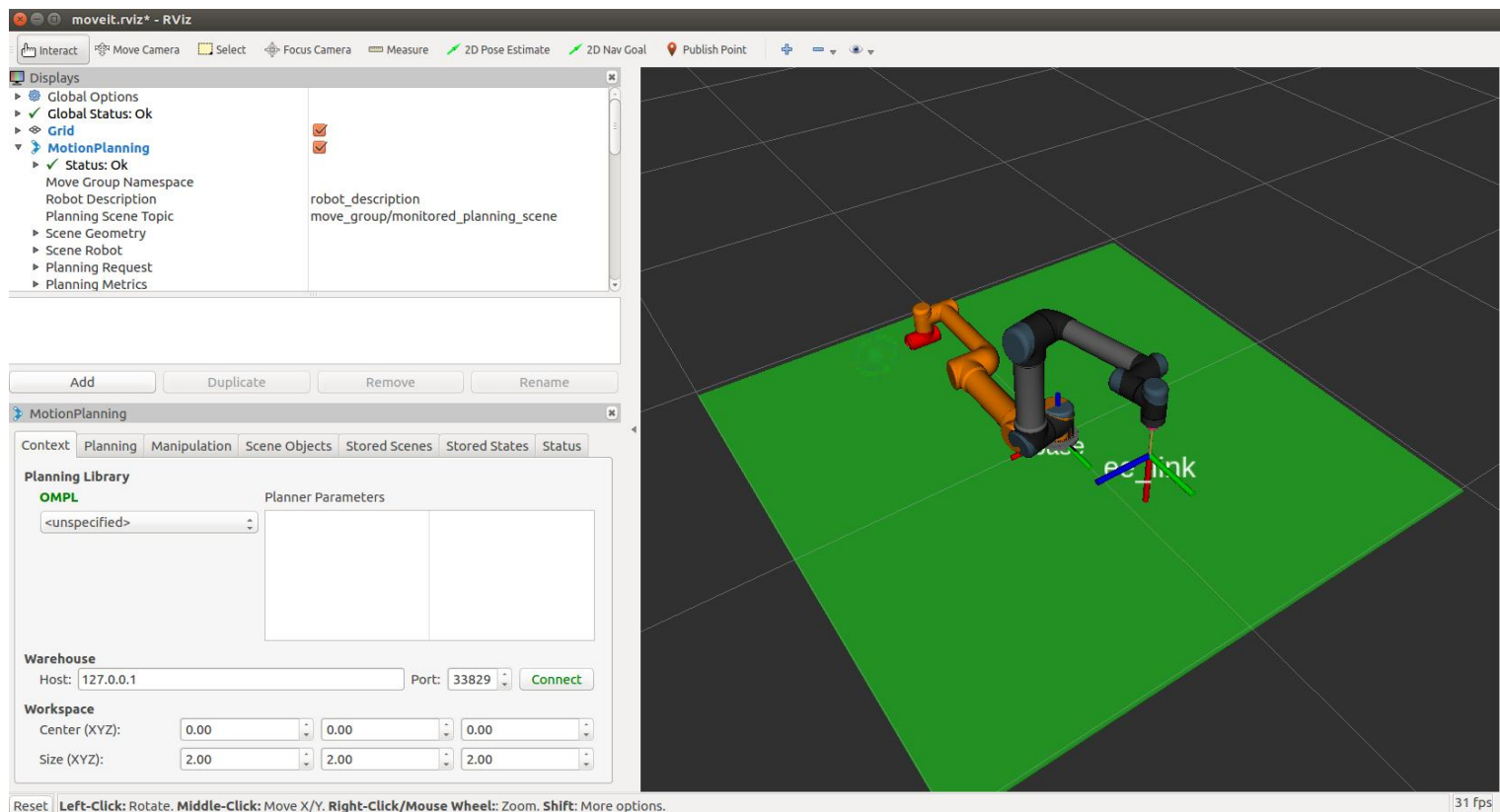
- 「₁」 Moveit!回顾
- 「₂」 关节空间运动
- 「₃」 笛卡尔空间运动
- 「₄」 与环境交互

■ 用户接口



Moveit!的核心节点——move_group

■ 用户接口



B站私信up: “002”

链接: https://pan.baidu.com/s/1leVPr4M9k_ayJMQ-RA-eFQ

提取码: 8888

■ move_j

```
bool move_j(const vector<double> &joint_group_positions) {  
    arm.setJointValueTarget(joint_group_positions);  
    arm.move();  
    sleep(0.5);  
    return true;  
}
```

2 关节空间运动

■ move_p

```
bool move_p(const vector<double> &pose) {

    //转换为moveit中pose的消息格式
    geometry_msgs::Pose target_pose;
    target_pose.position.x = pose[0];
    target_pose.position.y = pose[1];
    target_pose.position.z = pose[2];

    //RPY转换为四元数
    tf2::Quaternion myQuaternion;
    myQuaternion.setRPY(pose[3], pose[4], pose[5])
    target_pose.orientation.x = myQuaternion.getX();
    target_pose.orientation.y = myQuaternion.getY();
    target_pose.orientation.z = myQuaternion.getZ();
    target_pose.orientation.w = myQuaternion.getW();

    // 设置机械臂当前的状态作为运动初始状态
    arm.setStartStateToCurrentState();
    arm.setPoseTarget(target_pose);

    // 进行运动规划, 计算机器人移动到目标的运动轨迹, 此时只是计算出轨迹, 并不会控制机械臂运动
    moveit::planning_interface::MoveGroupInterface::Plan plan;
    moveit::planning_interface::MoveItErrorCode success = arm.plan(plan);

    ROS_INFO("Plan (pose goal) %s", success ? "" : "FAILED");

    //让机械臂按照规划的轨迹开始运动。
    if (success) {
        arm.execute(plan);
        sleep(1);
        return true;
    }
    return false;
}
```

■ 直线运动

```
bool move_1(const vector<double>& pose) {  
    vector<geometry_msgs::Pose> waypoints;  
    geometry_msgs::Pose target_pose;  
    target_pose.position.x = pose[0];  
    target_pose.position.y = pose[1];  
    target_pose.position.z = pose[2];
```

```
while (fraction < 1.0 && attempts < maxtries)  
{  
    fraction = arm.computeCartesianPath(waypoints, eef_step, jump_threshold, trajectory);  
    attempts++;  
}
```


■ 添加障碍物

```
void create_table() {  
  
    // Now let's define a collision object ROS message for the robot to avoid.  
  
    ros::Publisher planning_scene_diff_publisher = nh_.advertise<moveit_msgs::PlanningScene>("planning_scene", 1);  
    ros::WallDuration sleep_t(0.5);  
    while (planning_scene_diff_publisher.getNumSubscribers() < 1)  
    {  
        sleep_t.sleep();  
    }  
  
    moveit::planning_interface::PlanningSceneInterface planning_scene_interface;  
    moveit_msgs::PlanningScene planning_scene;  
    moveit_msgs::CollisionObject collision_object;  
    collision_object.header.frame_id = arm_->getPlanningFrame();  
  
    // The id of the object is used to identify it.  
    collision_object.id = "table";  
}
```


教程视频会持续更新

敬请期待！