# TIAGo Training Sessions

## Motion planning with MoveIt!

PAL ROBOTICS

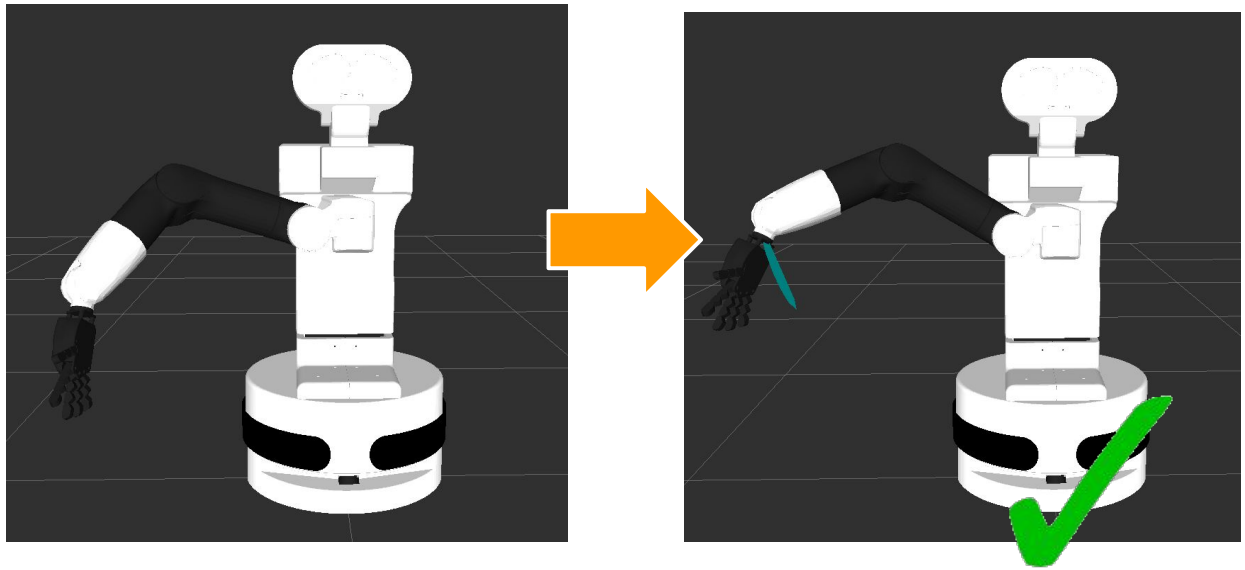
# Introduction

TIAGo T.S.

PAL ROBOTICS

# Movements without planning

- Pure spline interpolation for joint position
- No self-collision check
- Can be allowed in a controlled situation



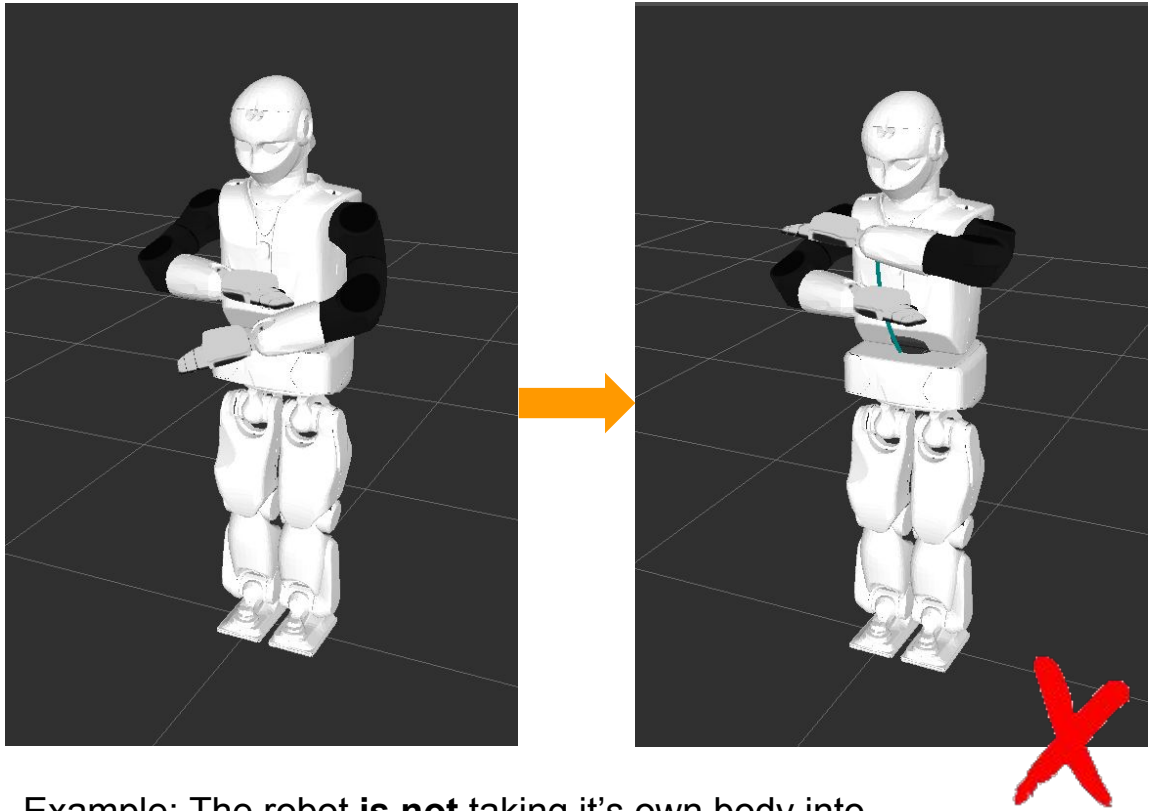


PAL ROBOTICS

# Collision avoidance (II)

Movements without planning

- Pure spline interpolation for joint position
- Can be allowed in a controlled situation
- Can lead to self collision

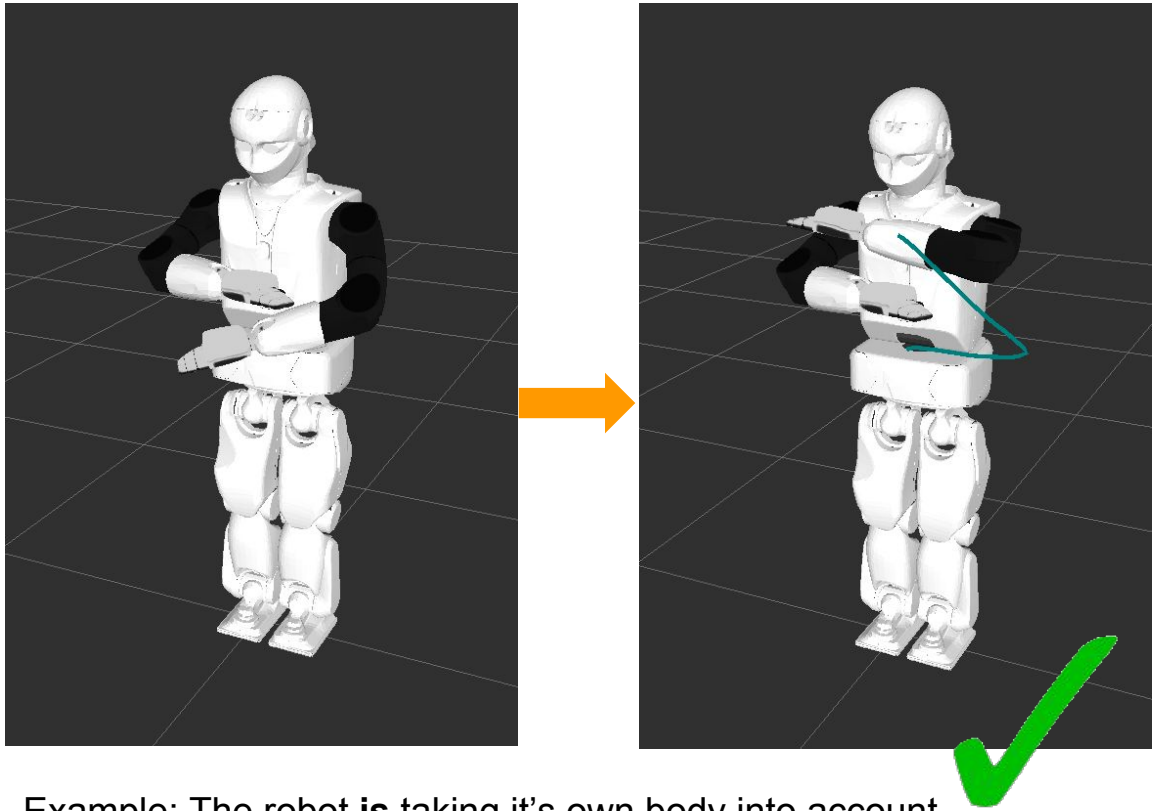


Example: The robot **is not** taking it's own body into account when executing motions.

# Collision avoidance (III)

“The robot is taking it’s own body into account when executing motions.”

How?



Example: The robot **is** taking it’s own body into account when executing motions.



PAL ROBOTICS

# Collision avoidance (IV)

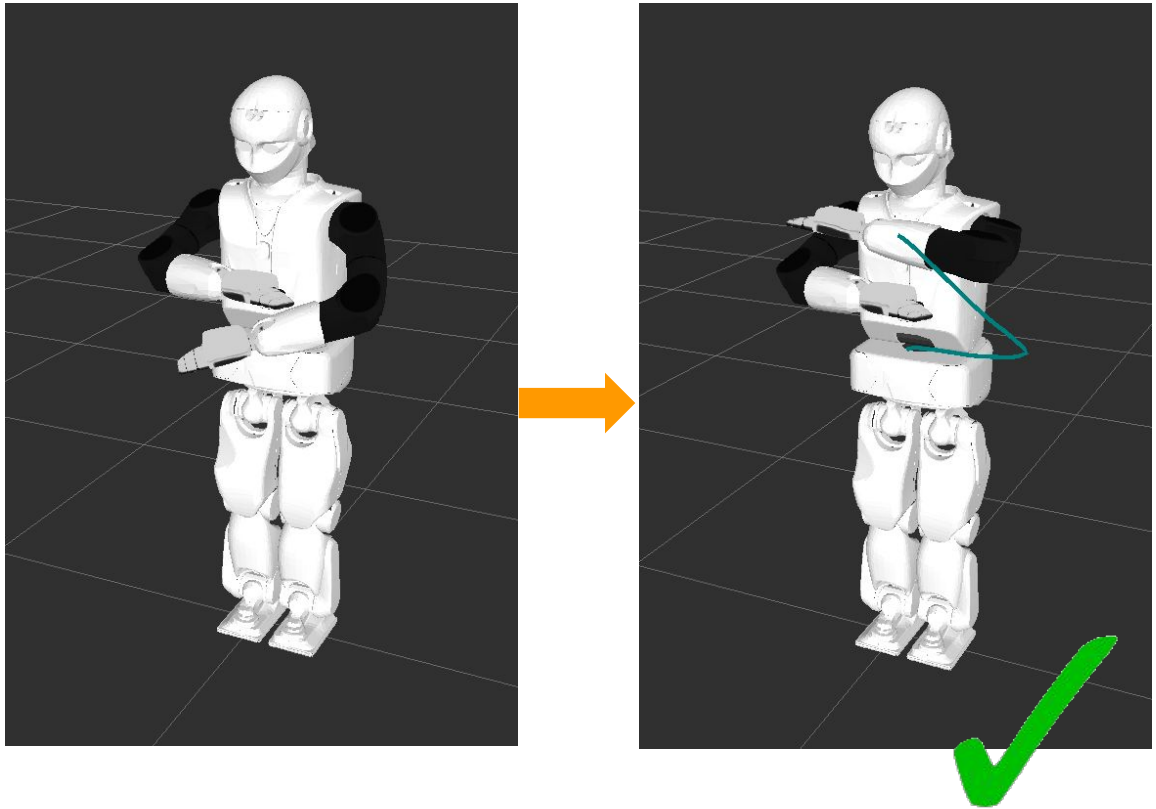
“The robot is taking it’s own body into account when executing motions.”

How?

Motion planning!

Probabilistic sampling-based motion planner: RRT-connect*

*MoveIt! has more planners available





PAL ROBOTICS

# MoveIt!

# MoveIt!

- “MoveIt! is state of the art software for mobile manipulation [...].”
- What does it provide for us?
  - Framework for motion planning and trajectory smoothing
    - ROS Action interface
    - Collision checking
    - Manage collision environment: add/remove (virtual) obstacles
  - Plugins for motion planning and trajectory smoothing
    - Change parameters of system
    - Develop custom plugins/algorithms
  - Tools to set up a robot configuration
  - GUI by means of an RViz plugin

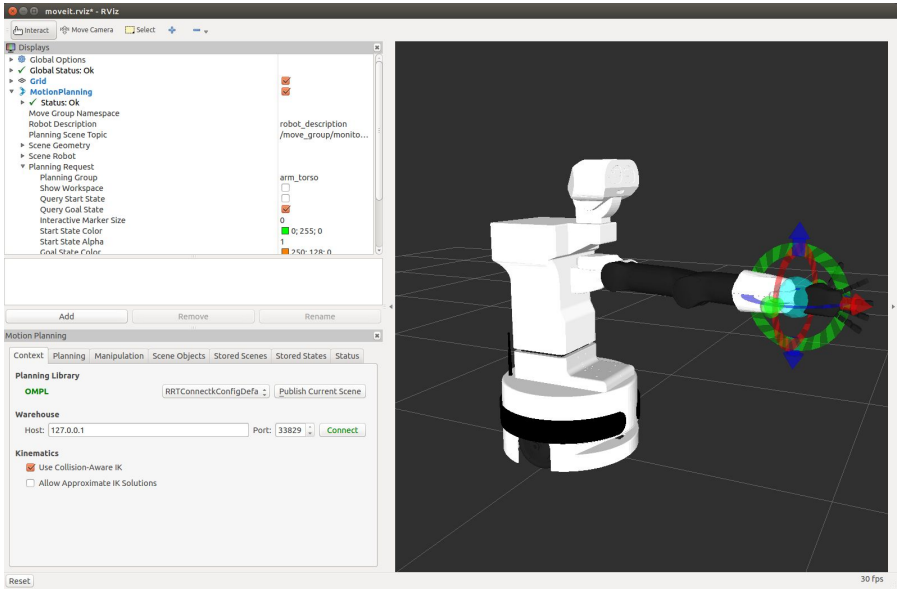


PAL ROBOTICS

MoveIt!

# Motion Planning GUI (I)

- Plugin of RViz
- Can be used with a real or simulated robot
- Supports visualization of plans without execution
- Select different planning groups
- On-the-fly change of planning library




Kinematic MoveIt! demo launch:

```
roslaunch tiago_moveit_config demo.launch
```

Real robot*:

```
export ROS_IP=10.68.0.128
export ROS_MASTER_URI=http://tiago-0c:11311
roslaunch tiago_moveit_config moveit_rviz.launch config:=true
```

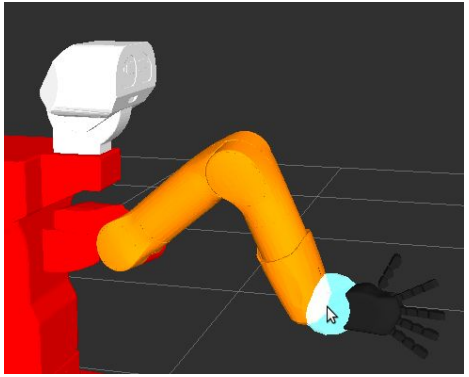
PAL ROBOTICS

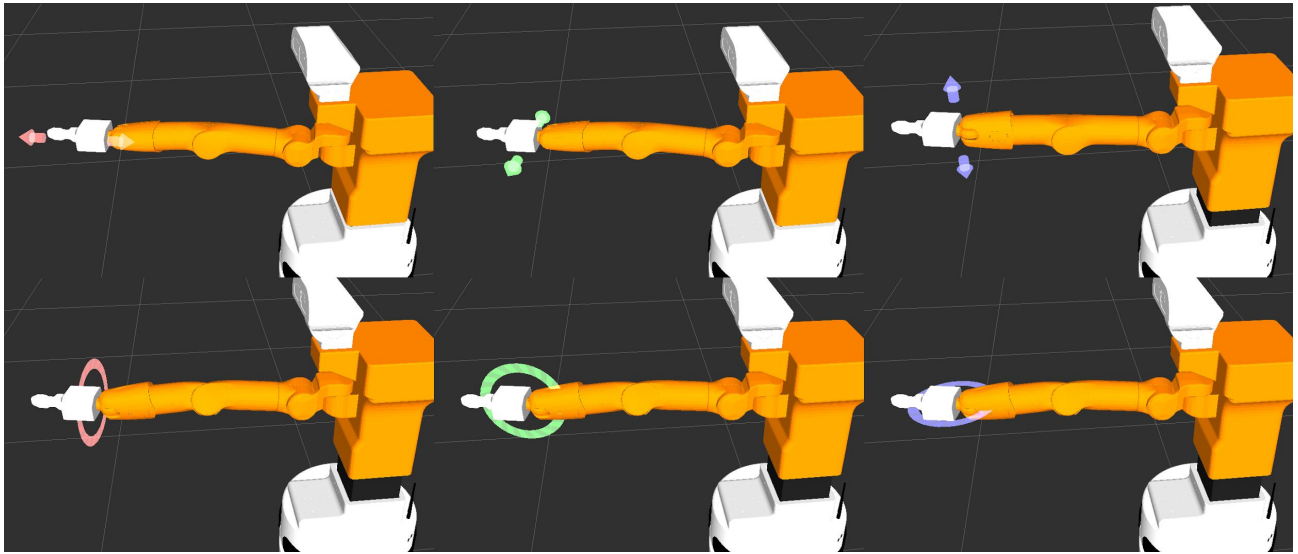
MoveIt!

TIAGo T.S.

# Motion Planning GUI (II)

- To specify the end effector goal,
- drag the end effector
  - along translational and
  - along rotational axis
  - in 3D





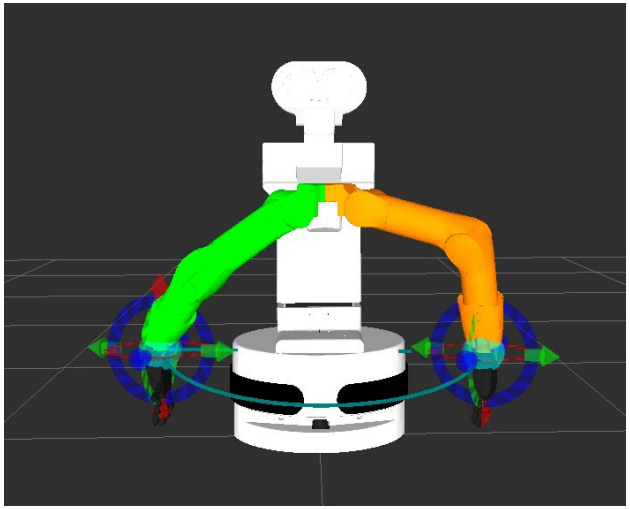


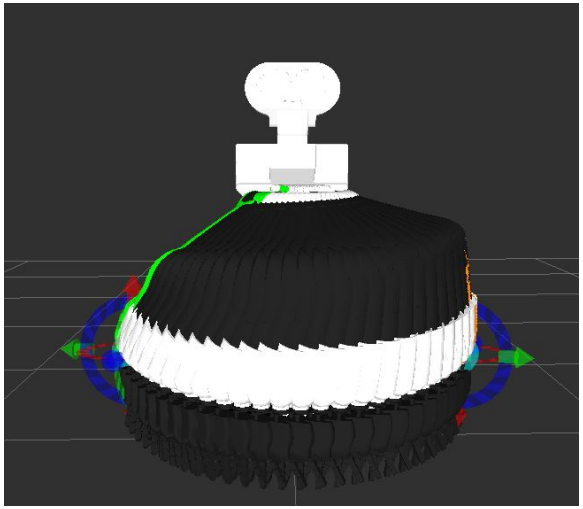
PAL ROBOTICS

MoveIt!

# Motion Planning GUI (III)

- When the goal state is not in collision, motion planning will generate trajectories that go around
  - robot body parts





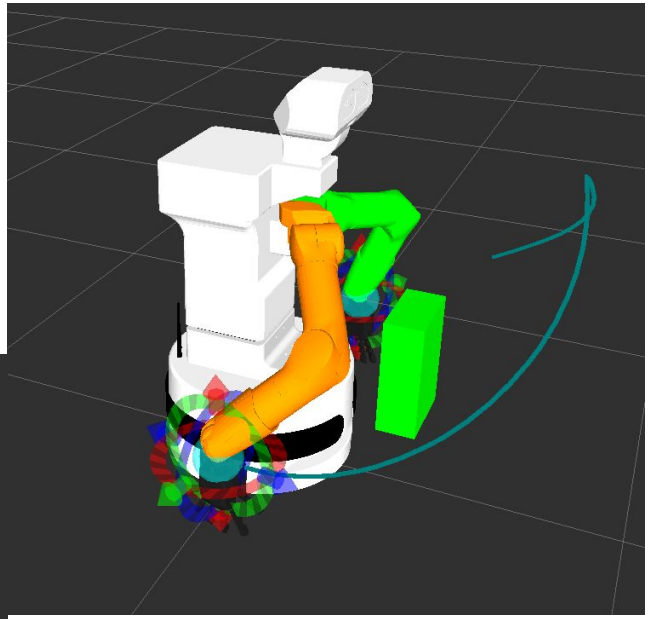


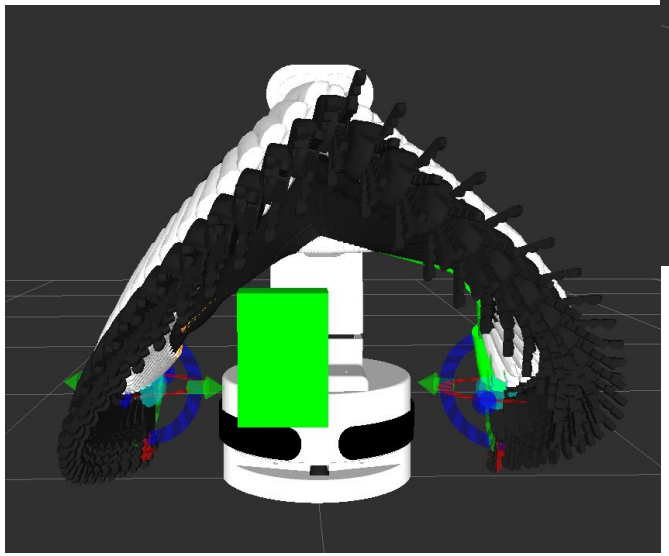
PAL ROBOTICS

MoveIt!

# Motion Planning GUI (IV)

- When the goal state is not in collision, motion planning will generate trajectories that go around
  - robot body parts and
  - other objects of the collision environment.

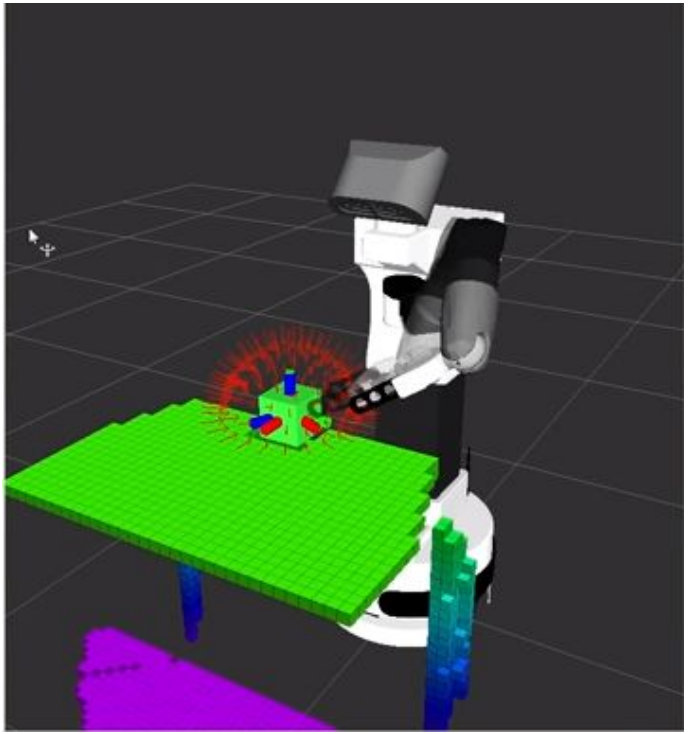




# Motion Planning GUI (V)

- When the goal state is not in collision, motion planning will generate trajectories that go around
  - robot body parts and
  - other objects of the collision environment. *

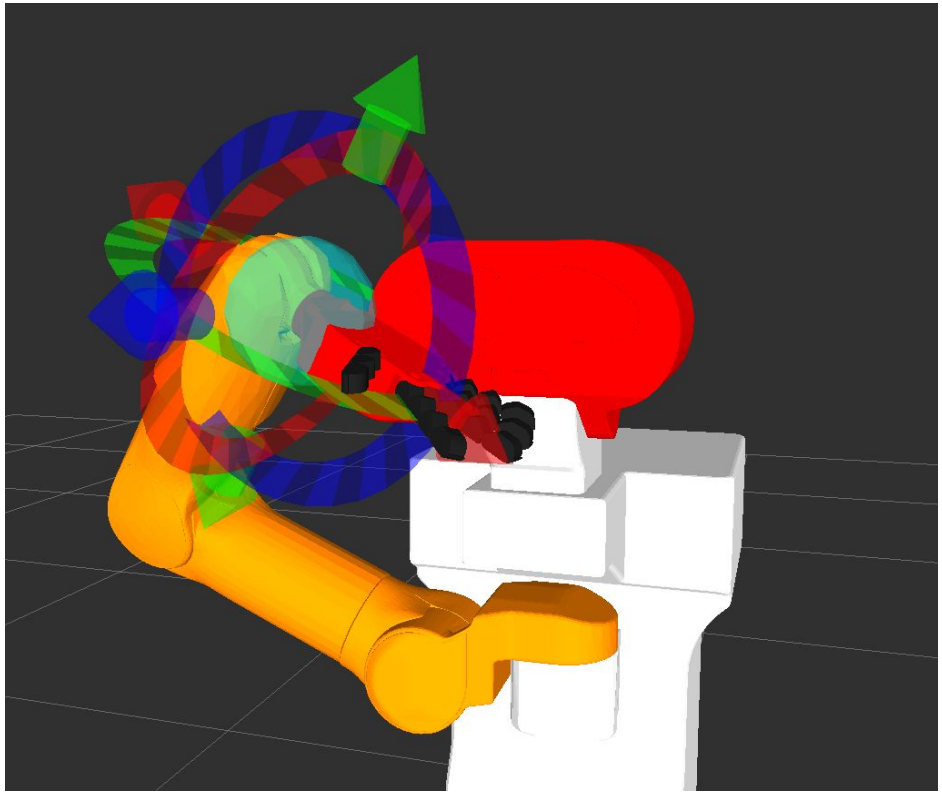
Objects can be added via the GUI, the action API or a **Kinect-type sensor (point clouds)**



# Motion Planning GUI (VI)

With a goal that creates collision,

- the GUI will draw colliding parts in red whether they are
  - self





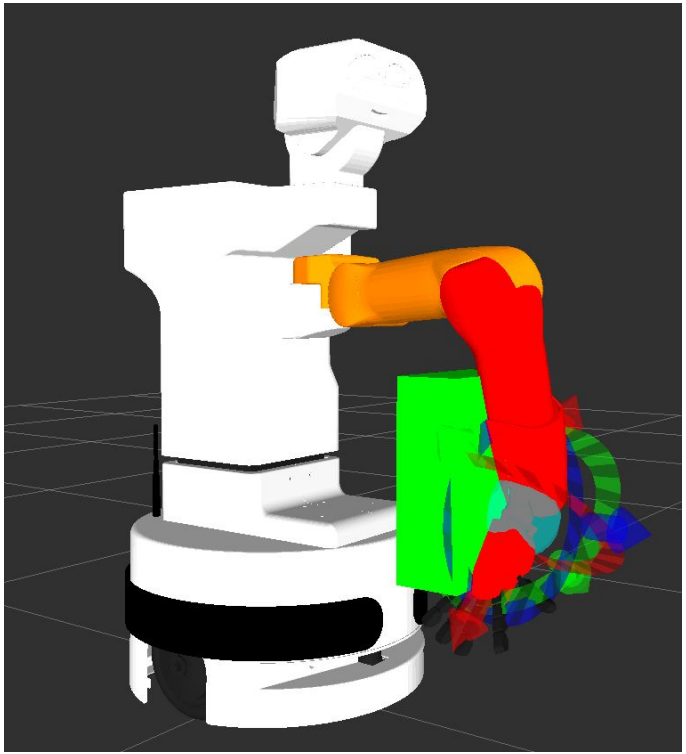
PAL ROBOTICS

MoveIt!

# Motion Planning GUI (VII)

With a goal that creates collision,

- the GUI will color colliding parts with red whether they are
  - self collisions
    or
  - environmental collisions.





PAL ROBOTICS

MoveIt!

# Learn more

- The official MoveIt! documentation can be found at: http://moveit.ros.org/documentation/
- With tutorials: http://moveit.ros.org/documentation/tutorials/
  - Basic C++:

http://docs.ros.org/indigo/api/pr2_moveit_tutorials/html/planning/src/doc/move_group_interface_tutorial.html

  - Basic Python:

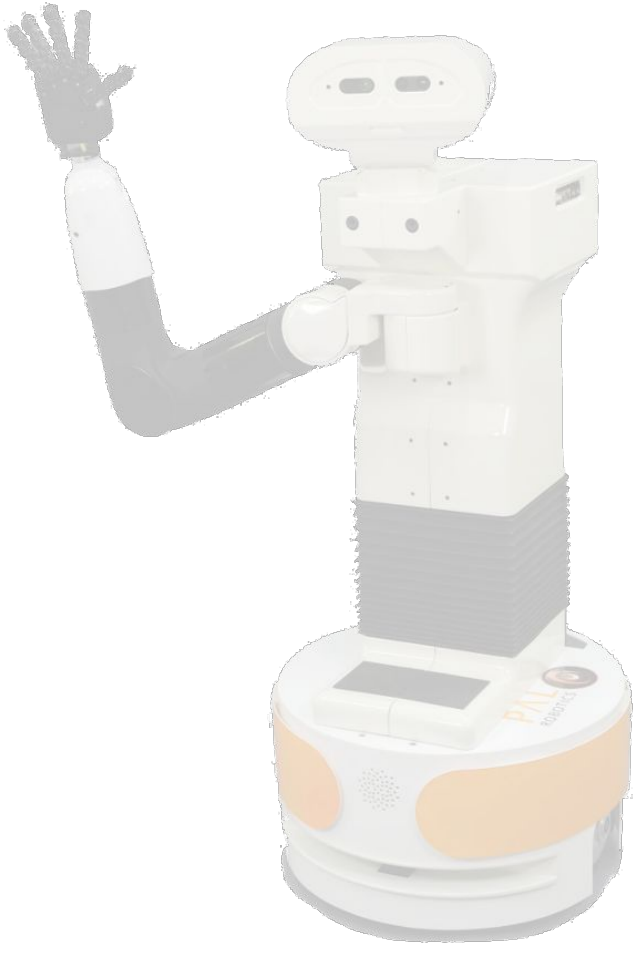
http://docs.ros.org/indigo/api/pr2_moveit_tutorials/html/planning/scripts/doc/move_group_python_interface_tutorial.html



PAL ROBOTICS

MoveIt!

# Questions?



PAL ROBOTICS