

# Lecture 1: Lecture Notes Building a simulation in ROS

## Object Manipulation and Task Planning

Victor Risager

February 6, 2024

## 1 Setup of ROS simulation

In order to grasp correctly, more advanced control of the gripper is needed. I believe he said impedance control.

## 2 Layers

### 2.1 Low level control

hardware

### 2.2 Movement Primitives

### 2.3 Robot Behaviour

### 2.4 Task planning

Putting the above together, gives task planning.

## 3 High Level Task Programming Robot Skills

Industrial manipulators are "inflexible", as they only do one thing. One way to program the robots to make them modular is to use Skillbased Programming. Large c++/python files → smaller modular (reusable) blocks of code.

As robotics engineers we usually design "robot primitives" but we need an intermediate layer between the task level, called "Robot Skills".

### 3.1 Robot Skill

A robot skill has an initial condition. it assesses the following questions.

- Can i do it?
- How am i doing?

Robot Skills are concatenated into skills sequences. The goal is to generate enough skill sequences, such that the operator can program on a task level.

Standard sequence of skills are programmed offline. Some parameterizations can be done online, e.g. the position of an object.

Kinesthetic teaching is a learning physically.

## 4 Gazebo

Gazebo has lazy loading, where it loads part of the map instead of the entire map at a time.

Remember to set mass and inertia parameters.

We need to provide collision geometry.

Inertia can also be calculated in meshlab.

max\_vel (xacro) is the correcting force that repels the gripper from penetrating the bounding object.

min\_depth is the minimum depth into the object before the repulsion force is triggered.

Robot controller effort is important, as e.g. distance controllers may not provide any effort to hold onto the object, therefore it does not grip the object.

### 4.1 Parameter testing

There is a testbed that can test the physics parameters in gazebo. RQT Dynamic Reconfigure.

### 4.2 URDF

Cannot do loops between links and joints. URDF is immutable when the robot\_description has been published. Nodes do not reread it.

### 4.3 Xacro

Provides loops. Xacro can provide sensor information. Reusable. It can import macros (reusable models (joint or links))

check\_urdf robot.urdf

Xacro has parameters that can be used for variables.

For mirrored object you can use "reflect parameters". This is done using

### 4.4 ROS2 Limitation

Conveyor and maybe the Computer vision part may have problems.