# Admittance Control in ROS2

## plus hints for command smoothing

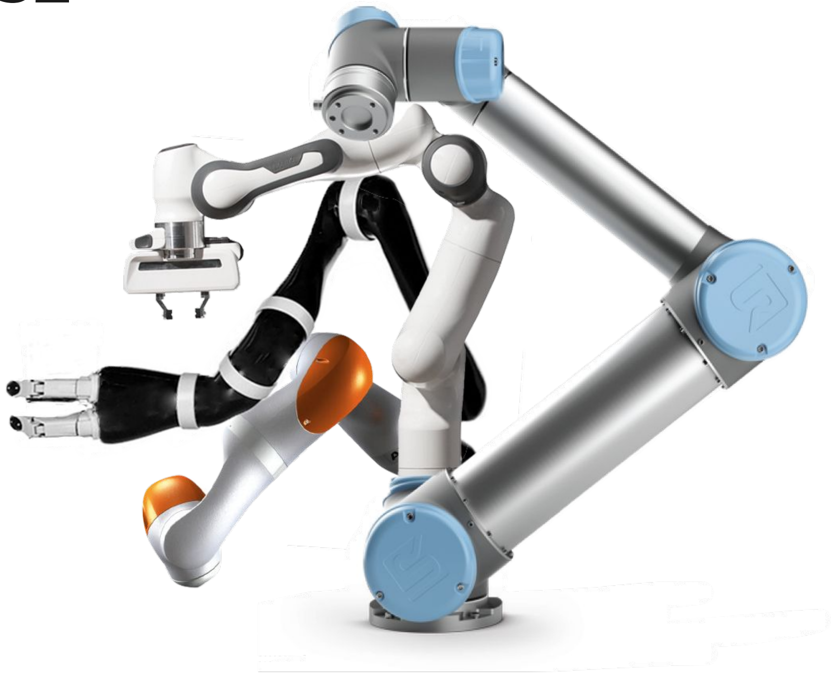
October 2021

**Dr. Andy Zelenak, Denis Stogl**
zelenak@picknik.ai, denis.stogl@picknik.ai

PICKNIK

# Admittance control in ROS2

- Motivation
- Video!
  - A real robot doing real work
- ros2_control implementation
- Block diagram
- How to use it
- Tips for motion smoothing



PICKNIK

# Admittance

What happens when a manipulator unexpectedly contacts the environment?

# Contact could occur because ...

- Dynamic environment
- Model or sensor uncertainty
- Manufacturing tasks require interaction with the physical world

% Nuclear Robotics Group

PICKNIK

# Contact with the environment

% Nuclear Robotics Group

PICKNIK

# A good way to handle this contact

- The robot acts like a spring
  - The robot moves in proportion to the measured wrench

Simplest example for one dimension:

$$\Delta x = \frac{1}{stiffness} \cdot F$$

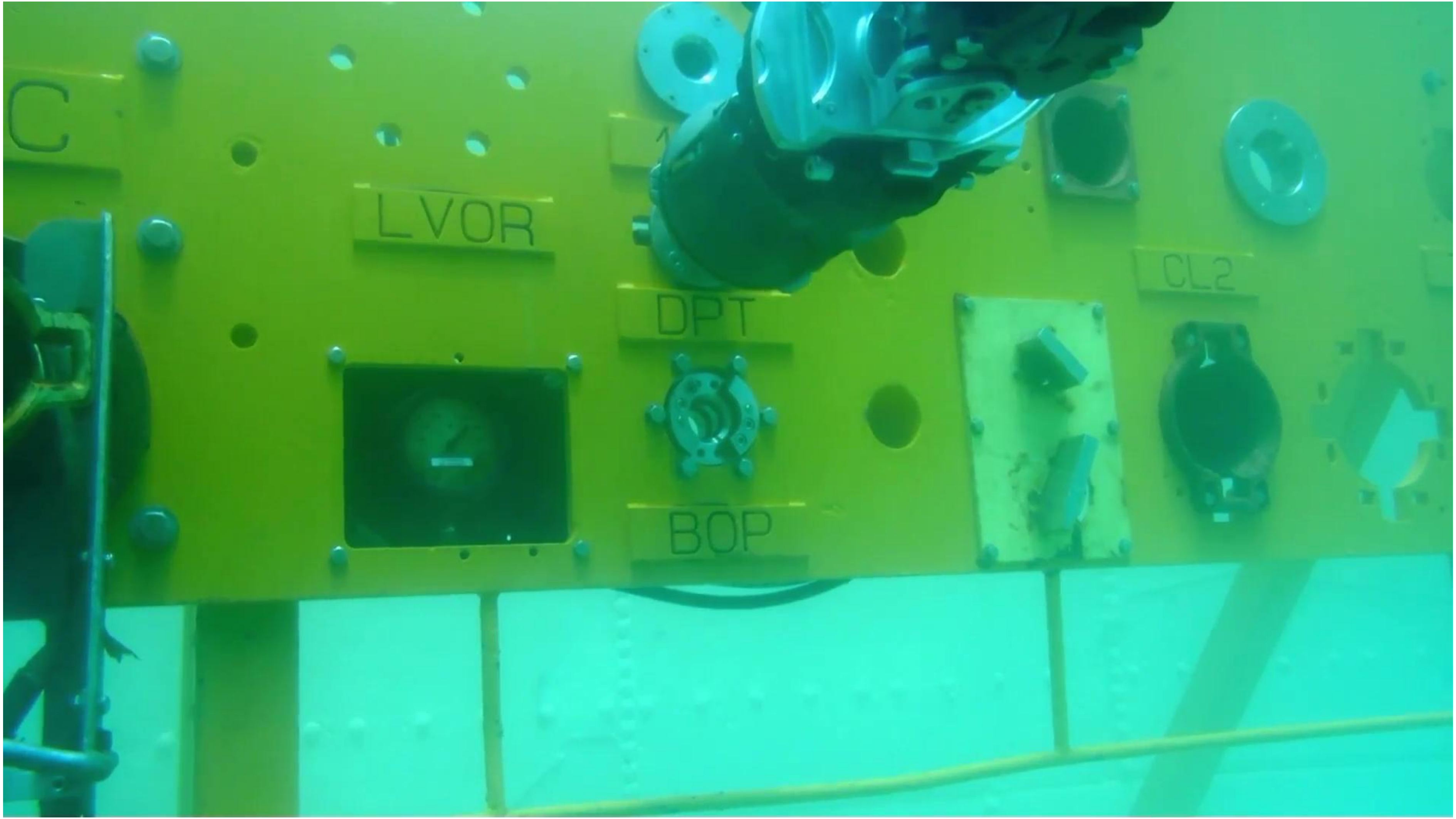
Full equation we used:

$$\ddot{x} = \frac{1}{mass} \cdot (F - damping \cdot \dot{x} - stiffness \cdot (x_{desired} - x_{current}))$$

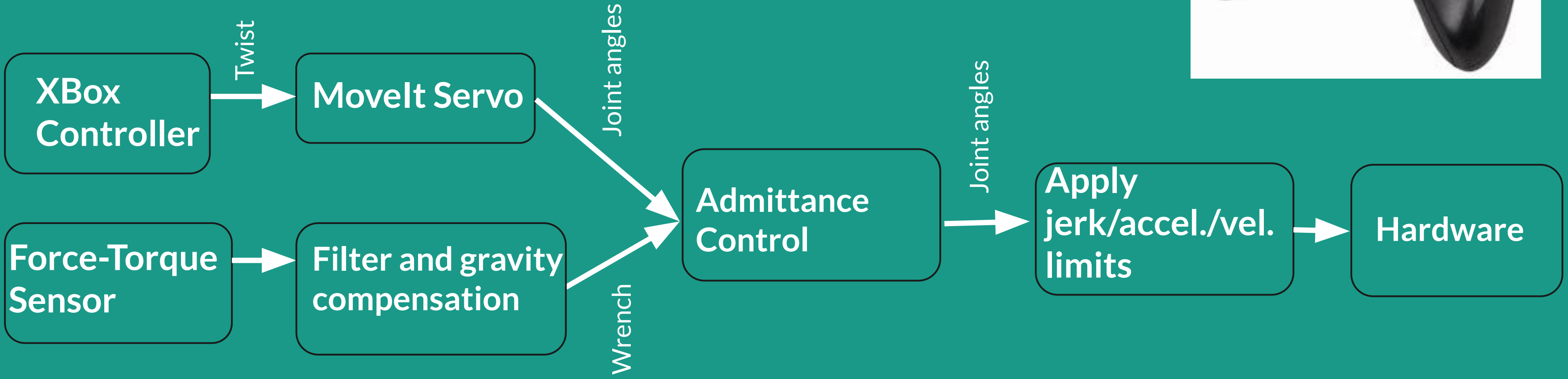
⟶ Mass/damping/stiffness are virtual parameters. Likely not equal to actual robot dynamics

⟶ Works for trajectories as well as online, streaming commands

# A large, powerful robot doing real work


C
LVOR
DPT
BOP
CL2

# Block Diagram



(Further refinements can be added. This is a good starting point)

# How to use it

PICKNIK

# Where to get it

- [Preliminary PR to ros2_control]
    - We expect to merge for the ROS2 Rolling distribution

- [The fully working branch]

PICKNIK

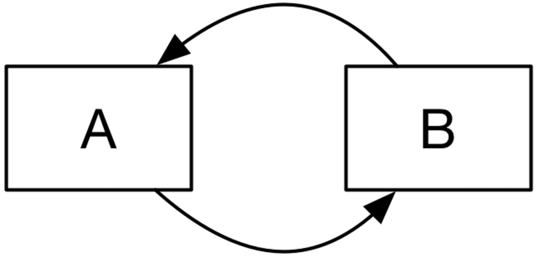
# Use it like any other ros2_controller

- Add ros2_control interfaces to ros2_control.xacro
  - Example
- Add controller name and parameters to controllers.yaml
  - Example
- Spawn the controller from a launch file
  - Example

# Circular Dependency Issue

- The draft PR depends on MoveIt2 to perform differential kinematics
  - We don't want a circular dependency between ros2_control and MoveIt2
  - But...
  - ros2_control probably should not get involved with robot kinematics

The solution may be a kinematics plugin to get the MoveIt dependency out of ros2_control



# Bonus -- Command Smoothing

- Reflexxes Type II in “velocity mode” works well for smoothing of streaming commands
  - Acceleration and velocity limits
  - A ROS-wrapped version (ROS1 and ROS2)
- Ruckig works well for smoothing of trajectories (i.e. multiple waypoints transmitted at once)
  - Velocity, acceleration, and jerk limits
  - MoveIt2 PR for inspiration
- Ruckig does not seem well-suited for streaming commands
- For stability: the output state of Reflexxes or Ruckig should be fed back as the current state of the robot in the next iteration