MujocoROS2Control: Seamless MuJoCo Integration with ROS 2 for Robot Simulation and Control

Adrian Danzglock[1] and Vamsi Krishna Origanti[1]

1 Robotics Innovation Center, German Research Center for Artificial Intelligence (DFKI), Bremen, Germany

Summary

The `MujocoROS2Control` hardware interface enables seamless integration between MuJoCo (Todorov, Erez, and Tassa 2012), a high-performance physics engine, and ROS 2 (ROS-Controls Team 2024a), a widely adopted middleware for robotic systems. This interface provides an efficient solution for simulating and controlling robots using MuJoCo's physics capabilities within the ROS 2 ecosystem.

To support ROS-based workflows, we developed a dedicated URDF-to-MJCF conversion script. This tool translates URDF models into MJCF (MuJoCo XML format), preserving kinematic and dynamic properties and allowing custom MuJoCo-specific parameters such as sensors, actuators, and collision definitions to be specified directly in the URDF. This conversion ensures compatibility and adaptability for simulation.

`MujocoROS2Control` bridges the gap between ROS 2 and MuJoCo, offering a streamlined workflow for simulation, controller testing, and reinforcement learning.

Statement of Need

Developing and validating control algorithms for robotic systems often requires extensive testing, which on physical hardware can be expensive, time-consuming, and subject to wear. Accurate simulation environments are essential for safe and scalable development.

MuJoCo (Multi-Joint dynamics with Contact) is a fast and accurate physics engine designed for robotics, control, biomechanics, and reinforcement learning. It provides: - Precise multi-body dynamics with advanced numerical integration, - High-speed simulation for real-time control and machine learning, - Sophisticated contact modeling and soft constraint handling, - Flexible actuator models and comprehensive sensor support.

Despite these capabilities, MuJoCo lacks native support for ROS 2, limiting its adoption in modern robotic development pipelines. `MujocoROS2Control` addresses this gap, enabling users to simulate ROS 2-compatible robots in MuJoCo with minimal overhead.

Our framework has been successfully used to test components such as force-torque sensor gravity compensation, torque-based Cartesian controllers, and Dynamic Movement Primitives (DMP)-based skill reproduction (Fabisch 2024).

Implementation

`MujocoROS2Control` integrates MuJoCo with the `ros2_control` framework. A key component is the URDF-to-MJCF converter, which maintains fixed joints (which MuJoCo typically

collapses), allows sensor and actuator tags within URDFs, and generates MJCF files compatible with MuJoCo's expectations.

The interface supports: - Direct torque control, - PID-based position/velocity/acceleration control, - MuJoCo's native actuator models.

Joint states and simulation time are published for synchronization with the ROS 2 system time (`/clock`). Sensors defined in the URDF (force-torque, IMU, pose, RGB-D camera) are exposed as individual ROS nodes using `realtime_tools` (ROS-Controls Team 2024b) to maintain real-time performance.

Examples

Franka FR3 with IndustRealKit Gears

This example integrates the Franka FR3 robot (FRANKA ROBOTICS 2024) with high-resolution gear models from the IndustRealKit (Tang et al. 2023). MuJoCo actuators are used to generate joint torques, although the implemented PID control and torque control are also supported. For the high-resolution collision modeling, we use CoaCD (Wei et al. 2022) to create multiple convex hulls from complex mesh geometry. The URDF-to-MJCF converter automatically replaces original mesh files in the mjcf, when the converted files are in the same directory.
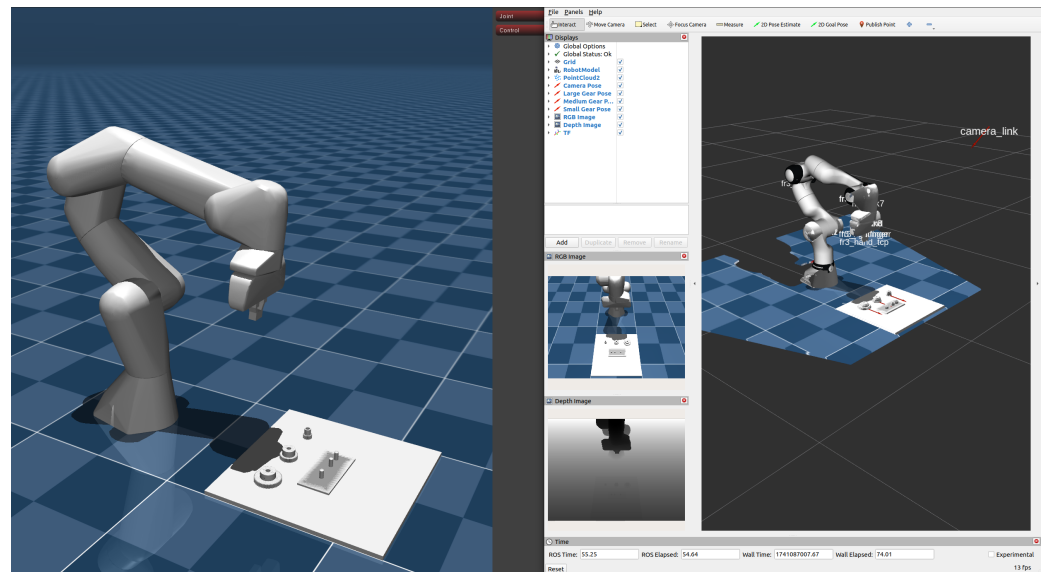
Figure 1: Franka FR3 controlled with ROS 2 Joint Trajectory Controller

IMRK System

In this example, we simulate the iMRK system developed at DFKI Bremen, consisting of two KUKA LBR iiwa 14 robots (**???**). A ROS2 Cartesian impedance controller (migrated version of (Mayr and Salt-Ducaju 2024)) is used for each arm. MuJoCo actuators manage the Robotiq 2F grippers, and force-torque sensors are simulated at both end-effectors.
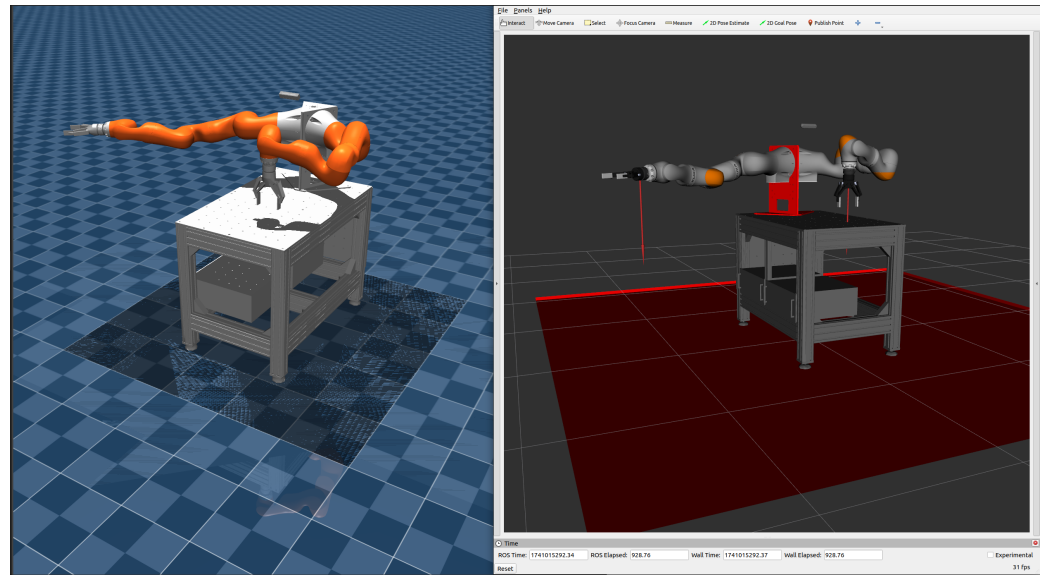
Figure 2: IMRK with Robotiq 2F Gripper and Robotiq FT300 Sensor

(Note: The IMRK robot description is internal and not included in the public examples.)

Conclusion

`MujocoROS2Control` enables high-fidelity robotic simulation by integrating MuJoCo with ROS 2. Through its robust conversion utilities, sensor bridging, and actuator support, it provides a reliable framework for control development, validation, and machine learning research.

Its lightweight design and fast simulation performance make it well-suited for force-based control, trajectory optimization, and reinforcement learning applications. Future enhancements may include support for additional sensor types, multi-robot coordination, and real-time closed-loop learning.

Acknowledgements

References

Fabisch, Alexander. 2024. *Journal of Open Source Software* 9 (97): 6695. https://doi.org/10.21105/joss.06695.

FRANKA ROBOTICS. 2024. https://github.com/frankaemika/franka_description.

Mayr, Matthias, and Julian M. Salt-Ducaju. 2024. "A C++ Implementation of a Cartesian Impedance Controller for Robotic Manipulators." *Journal of Open Source Software* 9 (93): 5194. https://doi.org/10.21105/joss.05194.

ROS-Controls Team. 2024a. https://github.com/ros-controls/ros2_control.

‌———. 2024b. https://github.com/ros-controls/realtime_tools.

Tang, Bingjie, Michael A Lin, Iretiayo Akinola, Ankur Handa, Gaurav S Sukhatme, Fabio Ramos, Dieter Fox, and Yashraj Narang. 2023. "IndustReal: Transferring Contact-Rich Assembly Tasks from Simulation to Reality." In *Robotics: Science and Systems*.

Todorov, Emanuel, Tom Erez, and Yuval Tassa. 2012. "MuJoCo: A Physics Engine for Model-Based Control." In *2012 Ieee/Rsj International Conference on Intelligent Robots and Systems*, 5026–33. IEEE. https://doi.org/10.1109/IROS.2012.6386109.

Wei, Xinyue, Minghua Liu, Zhan Ling, and Hao Su. 2022. "Approximate Convex Decomposition for 3d Meshes with Collision-Aware Concavity and Tree Search." *ACM Transactions on Graphics (TOG)* 41 (4): 1–18.