

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

Adrian Danzglock¹, Vamsi Krishna Origanti¹, and Frank Kirchner^{1,2}

¹ Robotics Innovation Center, German Research Center for Artificial Intelligence (DFKI), Bremen, Germany ² Universität Bremen, Bremen, Germany

DOI: [10.xxxxxx/draft](https://doi.org/10.xxxxxx/draft)

Software

- [Review](#)
- [Repository](#)
- [Archive](#)

Editor: [Open Journals](#)

Reviewers:

- [@openjournals](#)

Submitted: 01 January 1970

Published: unpublished

License

Authors of papers retain copyright and release the work under a Creative Commons Attribution 4.0 International License ([CC BY 4.0](#)).

Summary

The MujocoROS2Control hardware interface enables seamless integration between MuJoCo (Todorov et al., 2012), a high-performance physics engine, and ROS 2 (ROS-Controls Team, 2024b), 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.

MuJoCo is ideal for reliable torque feedback due to its high-fidelity torque-controlled joints and smooth-contact soft constraint solver, which enables precise compliance behavior (Zhang et al., 2025).

Gazebo, on the other hand, typically relies on hard-constraint engines like ODE or Bullet, which struggle with compliant torque-based tasks unless carefully tuned. These engines often require smaller timesteps than MuJoCo to maintain stability (Team, 2025).

Drake supports torque control, but due to its symbolic and rigid-body emphasis, tuning compliance can be computationally intensive (Tedrake & Drake Development Team, 2019).

Table 1: Comparison of usability of different Simulators for usage with Compliant controllers

Controller Type	MuJoCo	Gazebo/Ignition	Drake
Torque-based impedance	High-fidelity, fast, stable	Accurate but tuning required	Accurate, symbolic, slower than mujoco
Admittance with wrench input	Stable, smooth compliant	Plugin-lag, tuning required	precise, heavier computationally
Time-step flexibility	Larger stable steps (e.g. 2-5ms)	Smaller steps required (1ms)	Variable but slower dynamics

Table 2: Comparison of actual ros2 simulator wrappers

Feature	Mujoco ROS2 Control	mujoco ros2 control (MoveIt, 2025)	gz ros2 control (Controls, 2025)	drake-ros (Locomotion, 2025)
Simulation Engine	MuJoCo	MuJoCo	Gazebo	Drake
Sensor Support	Yes, IMU, Pose, Wrench, RGBD	No, Planned	Yes, Supports various sensors via Gazebo plugins	Yes, but additional code is required
URDF Support	Yes, direct loading from URDF via urdf to mjcf script in launchfile	No, Planned	Yes, Uses URDF/SDF for robot models	Yes, Supports URDF and custom formats
Control System	ros2_control	ros2_control	ros2_control	Experimental API for ROS2
Control Methods	PID, Mujoco Actuators, Torque	PID, Torque, not integrated position and velocity control	PID, Effort, Position, Velocity	PID, Optimazation-based-control
Mimic Joints	Yes	Yes	Yes, sometimes difficult to setup (User, 2025)	Yes

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.

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.

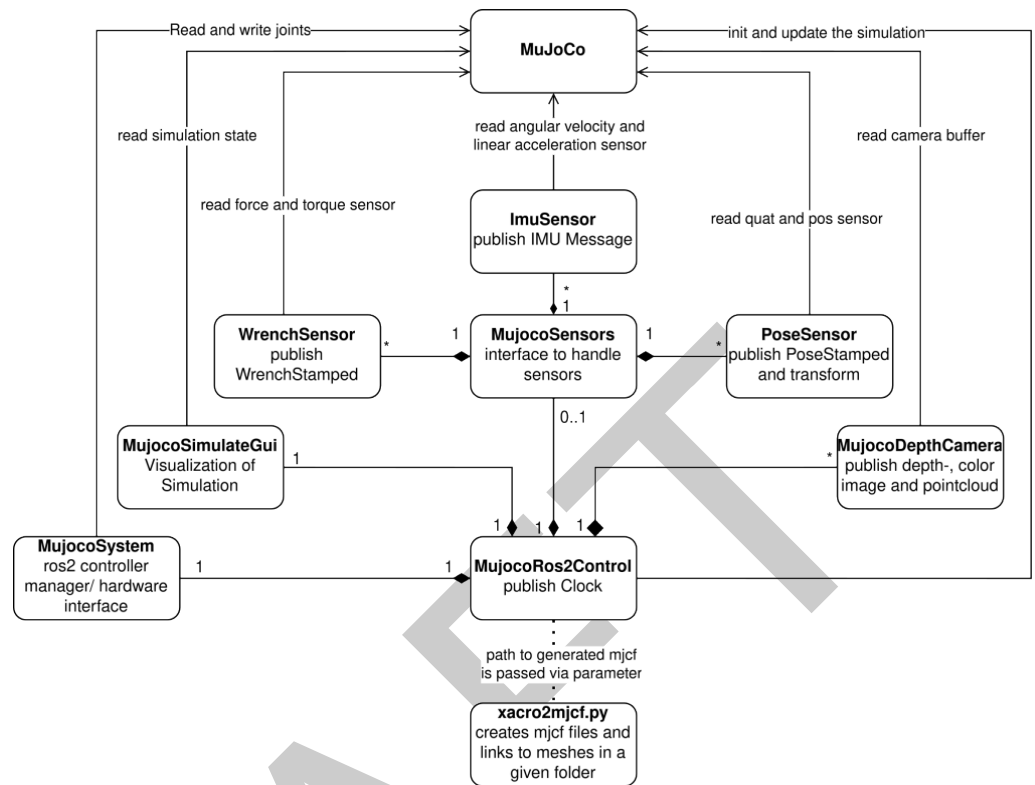


Figure 1: Overview of the MujocoRos2Control Structure

The interface supports direct torque control, PID-based position/velocity/acceleration control and 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 are exposed as individual ROS nodes using `realtime_tools` (ROS-Controls Team, 2024a) to maintain real-time performance.

Use Cases

MujocoRos2Control was utilized for testing and validating various torque and admittance controllers within the scope of the HARTU project (HARTU Project, n.d.). The software also played a key role in conducting experiments for the publication "Look-Ahead Optimization for Managing Nullspace in Cartesian Impedance Control of Dual-Arm Robots" (Origanti et al., 2025).

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

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

64 create multiple convex hulls from complex mesh geometry. The URDF-to-MJCF converter
65 automatically replaces original mesh files in the mjcf, when the converted files are in the same
66 directory.

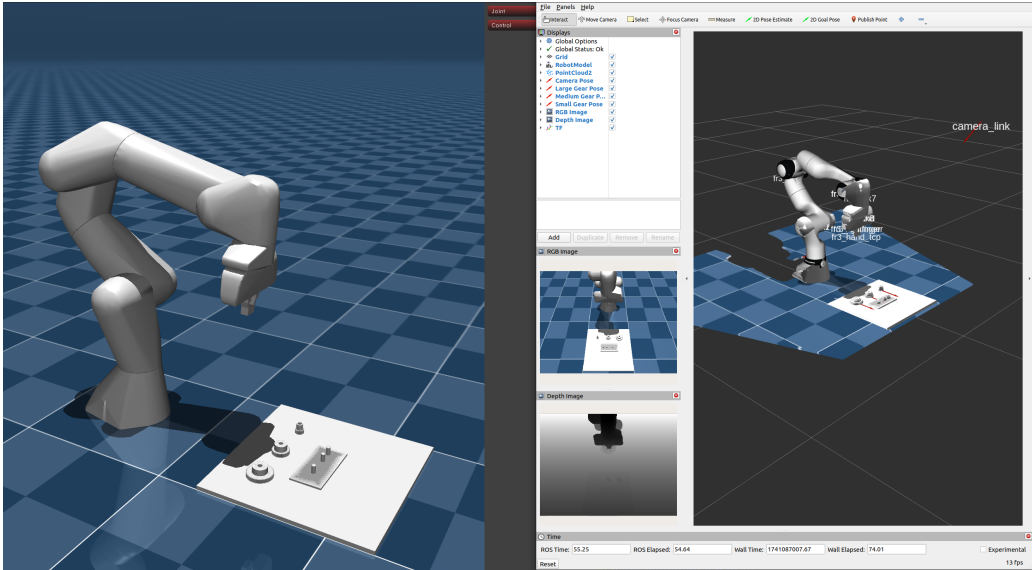
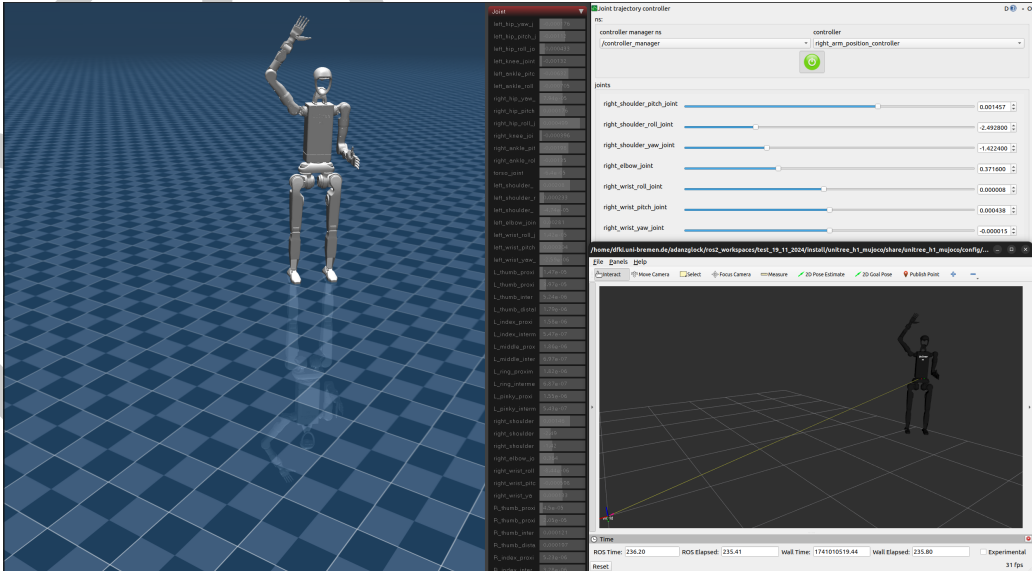


Figure 2: Franka FR3 controlled with ROS 2 Joint Trajectory Controller

67 **Unitree H1**

68 This example uses the loads the Bipedal robot Unitree H1 (Robotics, 2025) with a
69 floating base and tf2 transformations from world to pelvis. All joints are controlled via
70 ros2 control, with mujoco actuators for position and velocity control (PD Control). The



72 **iMRK System**

73 In this example, we simulate the iMRK system developed at DFKI Bremen, consisting of two
74 KUKA LBR iiwa 14 robots (Dennis Mronga, 2025). A ROS2 Cartesian impedance controller
75 (migrated version of (Mayr & Salt-Ducaju, 2024)) is used for each arm. MuJoCo actuators
76 manage the Robotiq 2F grippers, and force-torque sensors are simulated at both end-effectors.

77 Because the robot description for the IMRK system is not public available, we cant provide
78 the files for the examples.

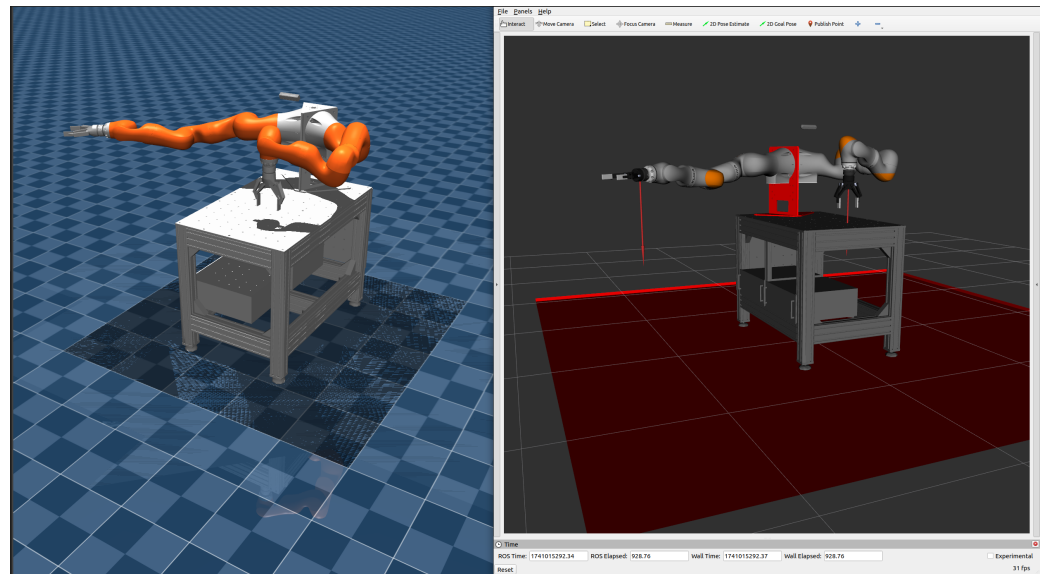


Figure 3: IMRK with Robotiq 2F Gripper and Robotiq FT300 Sensor

79 Conclusion

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

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

87 Acknowledgements

88 This library was initiated and developed at the Robotics Innovation Center, German Research
89 Center for Artificial Intelligence (DFKI GmbH), Bremen, Germany, as part of the HARTU
90 Project. This project received funding from the European Union's Horizon Europe research
91 and innovation program under grant agreement No. 101092100.

92 References

- 93 Controls, R. (2025). *Gazebo ROS2 control*. https://github.com/ros-controls/gz_ros2_control
- 94 Dennis Mronga. (2025). *iMRK dual-arm robot*. <https://robotik.dfki-bremen.de/en/research/robot-systems/imrk/>.
- 95
- 96 Fabisch, A. (2024). Movement_primitives: Imitation learning of cartesian motion with
97 movement primitives. *Journal of Open Source Software*, 9(97), 6695. <https://doi.org/10.21105/joss.06695>
- 98
- 99 FRANKA ROBOTICS. (2024). *Franka_description: Official models of franka robotics GmbH*
100 *robots*. https://github.com/frankaemika/franka_description.

- 101 HARTU project. (n.d.). <https://www.hartu-project.eu/>.
- 102 Locomotion, R. (2025). *Drake ROS integration*. [https://github.com/RobotLocomotion/](https://github.com/RobotLocomotion/drake-ros)
103 [drake-ros](https://github.com/RobotLocomotion/drake-ros)
- 104 Mayr, M., & Salt-Ducaju, J. M. (2024). A c++ implementation of a cartesian impedance
105 controller for robotic manipulators. *Journal of Open Source Software*, 9(93), 5194.
106 <https://doi.org/10.21105/joss.05194>
- 107 MoveIt. (2025). *MuJoCo ROS2 control*. https://github.com/moveit/mujoco_ros2_control
- 108 Origanti, V. K., Danzglock, A., & Kirchner, F. (2025). Look ahead optimization for managing
109 nullspace in cartesian impedance control of dual-arm robots. *2025 IEEE/SICE International*
110 *Symposium on System Integration (SII)*, 990–997. [https://doi.org/10.1109/SII59315.2025.](https://doi.org/10.1109/SII59315.2025.10871056)
111 [10871056](https://doi.org/10.1109/SII59315.2025.10871056)
- 112 Robotics, U. (2025). *Unitree_ros*. [https://raw.githubusercontent.com/unitreerobotics/](https://raw.githubusercontent.com/unitreerobotics/unitree_ros)
113 [unitree_ros](https://raw.githubusercontent.com/unitreerobotics/unitree_ros).
- 114 ROS-Controls Team. (2024a). *Realtime_tools: Contains a set of tools that can be used*
115 *from a hard realtime thread, without breaking the realtime behavior*. [https://github.com/](https://github.com/ros-controls/realtime_tools)
116 [ros-controls/realtime_tools](https://github.com/ros-controls/realtime_tools).
- 117 ROS-Controls Team. (2024b). *ros2_control: A generic and flexible control framework for*
118 *ROS 2*. https://github.com/ros-controls/ros2_control.
- 119 Tang, B., Lin, M. A., Akinola, I., Handa, A., Sukhatme, G. S., Ramos, F., Fox, D., & Narang,
120 Y. (2023). IndustReal: Transferring contact-rich assembly tasks from simulation to reality.
121 *Robotics: Science and Systems*.
- 122 Team, G. S. (2025). *Gazebo physics parameters*. [https://classic.gazebosim.org/tutorials?tut=](https://classic.gazebosim.org/tutorials?tut=physics_params&cat=physics)
123 [physics_params&cat=physics](https://classic.gazebosim.org/tutorials?tut=physics_params&cat=physics)
- 124 Tedrake, R., & Drake Development Team, the. (2019). *Drake: Model-based design and*
125 *verification for robotics*. <https://drake.mit.edu>
- 126 Todorov, E., Erez, T., & Tassa, Y. (2012). MuJoCo: A physics engine for model-based control.
127 *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 5026–5033.
128 <https://doi.org/10.1109/IROS.2012.6386109>
- 129 User, S. E. (2025). *Gazebo ROS2 control mimic joints not working as expected*. [https://](https://robotics.stackexchange.com/questions/110622/gazebo-ros2-control-mimic-joints-not-working-as-exp)
130 robotics.stackexchange.com/questions/110622/gazebo-ros2-control-mimic-joints-not-working-as-exp
- 131 Wei, X., Liu, M., Ling, Z., & Su, H. (2022). Approximate convex decomposition for 3d meshes
132 with collision-aware concavity and tree search. *ACM Transactions on Graphics (TOG)*,
133 *41(4)*, 1–18.
- 134 Zhang, J. Z., Howell, T. A., Yi, Z., Pan, C., Shi, G., Qu, G., Erez, T., Tassa, Y., &
135 Manchester, Z. (2025). *Whole-body model-predictive control of legged robots with*
136 *MuJoCo*. <https://arxiv.org/abs/2503.04613>