The iMETRO Dynamic Simulation: An Open-Source Simulator for Intravehicular Space Robotics Research

Nikki Hart[1,2], Nathan Dunkelberger[2,3], Erik Holum[2,4], Lydia E. Kavraki[1,5], Emma Zemler[2], Shaun Azimi[2]
[1]Rice University [2]NASA Johnson Space Center [3]CACI International [4]METECS [5]Ken Kennedy Institute

***Abstract*— We present the iMETRO Dynamic Simulation, the first open-source dynamic simulation environment for research in the use of robot manipulators inside space vehicles for maintenance and logistics tasks, or intravehicular robotics (IVR). IVR has great potential to facilitate science and exploration on the Moon by saving crew time, but there are limited open-source resources that would enable researchers to identify the next set of challenges in manipulation for IVR. We provide a full-featured, high-fidelity dynamic simulation of the real-world iMETRO IVR test facility, which includes mockups representative of the interior of a future space vehicle as well as an 8-DoF manipulator that serves as an example robot platform for this research. Our modular simulator enables new software, hardware, and operational paradigms to be tested in a reconfigurable mockup environment. To improve the accessibility and extensibility of this simulation environment, we also provide ROS 2 hardware control interfaces to MuJoCo as well as a model conversion tool such that the same models may be used with ROS 2 and MuJoCo. To evaluate the sim-to-real transfer capabilities of this simulation, we present an open-source example application demonstration developed in the simulation and transfer it to the real-world iMETRO facility in less than a day. Finally, we identify the challenges and opportunities in modeling a real-world facility to aid future simulation efforts. The open-source simulation and application can be found at `https://github.com/NASA-JSC-Robotics`. The MuJoCo and ROS 2 integration tools have migrated to the ros-controls organization and can be found at `https://github.com/ros-controls/mujoco_ros2_control`. NOTE: This is an abridged version of a full paper accepted to ICRA 2026.**

I. Introduction

As NASA plans to return humans to the Lunar surface by 2028, the use of robots for surface and orbital maintenance and logistics tasks has emerged as a promising way to save crew time and facilitate exploration objectives [1]. In particular, intravehicular robotics (IVR), or the use of robots inside space vehicles and habitats, has the potential to improve mission outcomes leveraging existing technologies. Unlike the algorithms required for extravehicular robotics (EVR), those required for IVR can assume stable lighting conditions, reliable access to sensing and power, and a relatively well-known environment. Similarly, hardware for IVR need not consider certain environmental conditions of space, such as lunar dust or large temperature fluctuations. IVR research is key to identifying the next set of challenges for robots to advance mission objectives in space. NASA has recently made efforts to open-source aspects of real or realistic intravehicular environments [2, 3] to facilitate research in this area, but further effort is required to create a realistic, publicly-accessible testbed for a breadth of IVR research.

To fill the gap of open-source dynamic simulations, which model sensor observations and contact dynamics in addition to a robot's proprioception, for intravehicular robotics research, we introduce the iMETRO Dynamic Simulation. Based on NASA's open-source iMETRO facility resources [3], our dynamic simulation provides a modular simulation environment in which new robotics software, platforms, and operational paradigms can be evaluated for IVR use cases. Our simulation includes a high-fidelity model of an 8-DoF robot manipulator; ROS 2 control hardware interfaces for the simulated robot and its sensors; and modular, high-fidelity models of real-world space vehicle mockups. We also provide an open-source application demonstration of a logistics task using the 8-DoF manipulator which is used to test sim-to-real transfer. In summary, our contributions are as follows:

- We present the first open-source dynamic simulation environment for robot manipulation research in IVR applications;
- we provide tools for MuJoCo and ROS 2 integration that facilitate the integration of new software and hardware components into the simulated environment;
- we demonstrate the sim-to-real transfer capabilities of the dynamic simulation environment through an example application demonstration;
- and we isolate and discuss the challenges we faced to aid future simulation efforts.

II. Related Work

A. Benchmarking and Simulation for Robot Manipulation

Driven by the rapid progression of algorithms and learned policies for robot manipulation, the robotics research community as a whole has identified the need for widely-accessible, standardized benchmarks for robot manipulation. High-fidelity dynamic simulations have emerged as the most common platforms for these benchmarks [4–12] as well as training arenas for new learned policies. Existing benchmarks and simulation environments are not suitable for IVR research, as the mechanisms and objects present in an IVR scenario vary greatly from the commercial off-the-shelf (COTS)

This work was supported by the NASA Johnson Space Center Engineering Innovation Fund. NH is supported by the Ken Kennedy Institute Computational Science and Engineering Recruiting Fellowship, funded by the Energy HPC Conference and the Rice University Department of Computer Science, and by the Rice University School of Engineering and Computing Graduate Fellowship. Work on this project by LEK is supported in part by NSF CCF-2336612 and Rice University Funds.

mechanisms and objects usually found in home or industrial environments. Namely, hardware found on a space vehicle is typically custom designed to meet spaceflight requirements such as space and weight saving or serve spaceflight-specific purposes such as cabin pressurization. Further, benchmarking environments for manipulation primarily focus on tabletop manipulation, while IVR operates on a larger scale.

B. Testbeds and Simulators for Intravehicular Robotics

Few suitable and accessible testbeds exist for IVR. Due to the unique conditions of space vehicle environments, tests of IVR systems often occur on the International Space Station (ISS) [13–16]. For earlier-stage research, NASA robotics researchers and their close collaborators often use realistic, large-scale ground mockups, such as the Granite Lab at Ames Research Center [17] or the iMETRO test facility at Johnson Space Center [3]. For most researchers, however, the cost or requirements for access to these resources can be prohibitive. Meanwhile, due to the lack of open-source real or realistic models of the inside of a space vehicle, there are few simulations of IVR environments. NASA and JAXA have both open-sourced IVR-focused ISS simulators: the Astrobee simulator [18] and the Int-Ball2 Simulator [19], both based in Gazebo [20]. While both simulators provide photo- and depth-realistic simulation of the ISS, the models are not articulated, and they therefore are not suitable testbeds for IVR manipulation. Open Robotics provides a Gazebo model of the ISS [21], including graspable handles, but there are no large-scale articulated assets for testing IVR logistics and maintenance tasks. Finally, NASA has also open-sourced models of the aforementioned iMETRO facility [3] but only simulate robot kinematics. Therefore, contact dynamics and uncertainty are not considered for manipulation applications, and sensor data cannot be simulated to represent the state of the simulated world.

III. Design of the iMETRO Dynamic Simulation

A. MuJoCo and ROS 2 Integration Tools

We build the iMETRO Dynamic Simulation off of MuJoCo [12], an open-source physics simulator with a strong focus on contact dynamics and real-time performance on CPUs, facilitating our goal of providing realistic simulation for manipulation while also remaining extensible for new applications. We extend MuJoCo's accessibility and features with a ROS 2 control hardware interface for MuJoCo, which provides expanded functionality compared to previous open-source implementations [22], including support for RGBD cameras, LiDAR, and force-torque sensors. Further, we provide a tool to convert unified robot description format (URDF) files, which are commonly used in ROS 2, to the MJCF file format used by MuJoCo. As a result, robots currently being used in ROS 2 can easily be integrated into the simulation.

B. Realistic Models of Real-world Mockups and Robot

The iMETRO Dynamic Simulation simulates the real-world space vehicle mockup environment found in the iMETRO facility. These mockups provide some of the unique mechanisms with which a robot may be expected to interact in an IVR scenario, including hatch handles, drawer pull mechanisms, a locking latch on the MERLIN freezer door, and hatch pressurization wheels. A mockup of a cargo transfer bag (CTB) is also included as a free body in the environment; while the CTB can deform slightly in the real world, it is modeled as a rigid object for simplicity. The default environment found in the simulation can be reconfigured using xacro [23] and our provided unified robot description format (URDF) to MuJoCo MJCF conversion tool. The mockups are assigned mass properties based on estimated material densities, and frictional properties are approximated based on visual inspection of the mockups' behavior in simulation relative to observations in the real world.

While the iMETRO facility is designed to be robot agnostic, the facility provides an example robot consisting of a UR10e manipulator on a Vention linear rail [24] and an Ewellix vertical lift kit [25], and we simulate these assets. The robot also features a Robotiq Hand-E parallel jaw gripper with custom fingers designed for IVR [26], a force-torque sensor located at the tool flange, and two RealSense cameras mounted at the wrist and vertical lift.

C. Open-source Application Demonstration

Finally, we provide an example application, including procedures to run the application in simulation and in the physical iMETRO facility. This example application highlights key features of the simulation, demonstrates a typical approach to robot behaviors in the iMETRO facility, and validates the sim-to-real capabilities of the dynamic simulation. The example application uses MoveIt [27] and other open-source software packages, making it extensible to new applications. The methods used to create the example application demonstration are described below.

IV. Application Demonstration Techniques

A. Planning and Perception

The example application demonstration is executed via a series of waypoints, including robot joint configurations, end-effector poses, and relative moves in the end-effector frame. We use the OMPL [28] planners in MoveIt for motion planning due to their integration with ROS 2. Both in simulation and on the real robot, we use a digital twin for motion planning. In simulation, the MuJoCo-to-ROS2-control hardware interface publishes joint states for these mockups onto ROS 2 topics, keeping the state of the digital twin up-to-date. When running the application demonstration on the real robot, we call a ROS 2 service within the application demonstration to update the world model's state once the bench seat open is complete, enabling safe motion planning.

We do not hard-code the waypoint for the CTB grasp; we instead identify a grasp point for the CTB by segmenting the image from the wrist-mounted RealSense camera into named colors [29] and identifying the centroid and depth of the red tape marking the CTB handle. The demonstration can therefore accommodate small variations in the CTB's location in the facility.

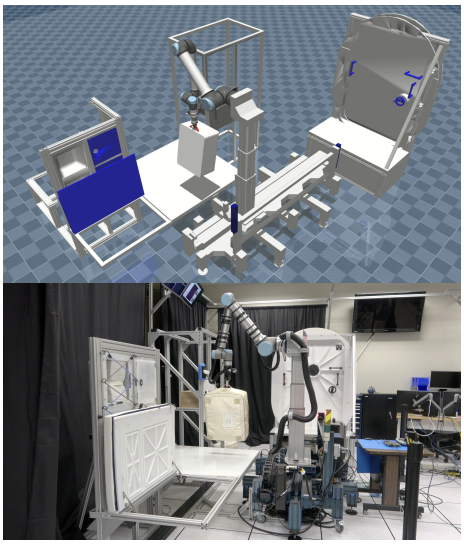

Fig. 1: Side-by-side images of the 8-DoF manipulator placing the CTB into the bench seat in simulation (top) and in the real facility (bottom).

B. Demonstration Procedure

The example application demonstrates a logistics task within the iMETRO facility using its 8-DoF manipulator. First, the robot approaches and opens the bench seat. The robot then traverses across the facility to approach the CTB. Using the color segmentation approach, the robot identifies a grasp pose on the CTB handle, grasps the CTB, and lifts the CTB. Finally, the robot carries the CTB to the bench seat and places the CTB upright in the seat's stowage area.

V. Results

To evaluate the sim-to-real transfer capabilities of the simulation, we transferred the example application demonstration described above to the real-world iMETRO facility, as shown in Figure 1. In this section, we describe the results of this experiment. Here, success for each step of the demonstration is defined as completing the action without unintended contact between the robot and its environment or in-hand objects and the environment, and overall task success is defined as the CTB being placed in an upright, stable position in the bench seat once the robot has retreated from the bench seat.

Prior to transferring the application demonstration to the real world, four trials were run in simulation with a task success rate of 100%. After the changes made to the simulation and behaviors as a result of real-world tests, two trials were run in the real-world iMETRO facility, and both were fully successful. To ensure changes made to the behaviors on the real robot did not impact task success in simulation, three additional trials of the adjusted demonstration were run in the updated simulation, and all were fully successful.

Transferring the example application demonstration built only in the iMETRO Dynamic Simulation was completed in less than a day. The sim-to-real transfer experiment of the application demonstration was used to identify modeling shortfalls, as described and categorized in Table I; however, we found that some of these shortfalls result from multiple contributing factors and are best mitigated through changes to the robot behaviors.

Demonstration Step	Failure Mode	Modeling Shortfall
Bench Seat Open	N/A	N/A
Traverse to CTB	E-stop due to robot's proximity to obstacles	Unmodeled camera cable housing
CTB Grasp	CTB handle out of camera frame	Unmodeled structural beams on floor
CTB Lift	Failure to plan	
CTB Place	CTB collision with mockups	CTB modeled as rigid

TABLE I: Failure modes that occurred during each demonstration step in the iMETRO facility and their associated modeling shortfalls.

VI. Lessons Learned

Identify changes to models. The iMETRO facility is a dynamic, complex, and modular environment; the robot, the mockups, and other assets in the facility may change at any time. Detecting changes to semi-structured environments is an open problem in IVR research, and perception-based methods for detecting discrepancies between a model and a real-world environment may be used for this purpose [17, 30].

Simulate uncertainty. In benchmarking literature, benchmarking problems are often procedurally generated with some degree of variation to evaluate generalization [4, 8, 31]. Extensions of this work should consider procedurally and reproducibly embedding uncertainty into the simulation environment, both to better simulate the real world and to improve the robustness of behaviors developed in simulation.

Maintain state of the world. Future work should consider methods of maintaining an up-to-date world state that lends itself to both geometric and symbolic representations, such as using AprilTags to identify both the position of a hatch door and whether it is fully open, partially open, or closed.

VII. Future Work

High-fidelity simulation of complex mechanisms. We provide a variety of complex mechanisms with which a robotic manipulator may interact, and in future work, we plan to evaluate the sim-to-real transfer of interactions with these mechanisms and to better model the behavior of complex features in the environment.

Quantitative evaluation of simulation fidelity. Quantitative methods must be used to fully characterize the properties of the robot and mockups in simulation and the real world. Force-torque sensor data, proprioceptive data, and proposed system identification tools [32] can be used to evaluate the fidelity of the robot and mockups.

Model additional robot embodiments and sensors. In ongoing work, we model a dual-arm mobile manipulator in the iMETRO Dynamic Simulation to explore simulation challenges such as localization, contact dynamics between mobile robots and the ground, and sensing modalities such as LiDAR and IMUs. Future work should leverage additional sensing modalities beyond the RGBD camera used in this work to identify the next set of simulation and sim-to-real transfer challenges.

REFERENCES

[1] National Aeronautics and Space Administration (NASA) Exploration Systems Development Mission Directorate (ESDMD). *Moon to Mars Architecture Definition Document (ESDMD-001)*. https://www.nasa.gov/wp-content/uploads/2024/12/esdmd-001-add-rev-b.pdf?emrc=5ffbf4. Version B.1. 2024.

[2] S. Kang, R. Soussan, D. Lee, B. Coltin, A. M. Vargas, M. Moreira, K. Browne, R. Garcia, M. Bualat, T. Smith, J. Barlow, J. Benavides, E. Jeong, and P. Kim. "Astrobee ISS Free-Flyer Datasets for Space Intra-Vehicular Robot Navigation Research". In: *IEEE Robotics and Automation Letters* 9.4 (2024), pp. 3307–3314.

[3] N. Dunkelberger, E. Sheetz, C. Rainen, J. Graf, N. Hart, E. Zemler, and S. Azimi. "Design of the iMETRO Facility: A Platform for Intravehicular Space Robotics Research". In: *2025 22nd International Conference on Ubiquitous Robots (UR)*. 2025, pp. 390–397.

[4] M. Heo, Y. Lee, D. Lee, and J. J. Lim. "FurnitureBench: Reproducible Real-World Benchmark for Long-Horizon Complex Manipulation". In: *Robotics: Science and Systems*. 2023.

[5] P. Atreya, K. Pertsch, T. Lee, M. J. Kim, A. Jain, A. Kuramshin, C. Eppner, C. Neary, E. Hu, F. Ramos, et al. "RoboArena: Distributed Real-World Evaluation of Generalist Robot Policies". In: *Proceedings of the Conference on Robot Learning (CoRL 2025)*. 2025.

[6] Y. Urakami, A. Hodgkinson, C. Carlin, R. Leu, L. Rigazio, and P. Abbeel. "Doorgym: A scalable door opening environment and baseline agent". In: *arXiv preprint arXiv:1908.01887*. 2019.

[7] Y. Zhu, J. Wong, A. Mandlekar, R. Martín-Martín, A. Joshi, K. Lin, S. Nasiriany, and Y. Zhu. "robosuite: A Modular Simulation Framework and Benchmark for Robot Learning". In: *arXiv preprint arXiv:2009.12293*. 2020.

[8] S. James, Z. Ma, D. R. Arrojo, and A. J. Davison. "RLBench: The Robot Learning Benchmark & Learning Environment". In: *IEEE Robotics and Automation Letters* 5.2 (2020), pp. 3019–3026.

[9] Y. Lee, E. S. Hu, and J. J. Lim. "IKEA Furniture Assembly Environment for Long-Horizon Complex Manipulation Tasks". In: *2021 IEEE International Conference on Robotics and Automation (ICRA)*. 2021, pp. 6343–6349.

[10] B. Liu, Y. Zhu, C. Gao, Y. Feng, Q. Liu, Y. Zhu, and P. Stone. "LIBERO: Benchmarking Knowledge Transfer for Lifelong Robot Learning". In: *Advances in Neural Information Processing Systems* 36 (2023), pp. 44776–44791.

[11] J. Gu, F. Xiang, X. Li, Z. Ling, X. Liu, T. Mu, Y. Tang, S. Tao, X. Wei, Y. Yao, X. Yuan, P. Xie, Z. Huang, R. Chen, and H. Su. "ManiSkill2: A Unified Benchmark for Generalizable Manipulation Skills". In: *International Conference on Learning Representations*. 2023.

[12] E. Todorov, T. Erez, and Y. Tassa. "MuJoCo: A physics engine for model-based control". In: *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE. 2012, pp. 5026–5033.

[13] M. Diftler, J. Mehling, M. Abdallah, N. Radford, L. Bridgwater, A. Sanders, R. Askew, D. Linn, J. Yamokoski, F. Permenter, B. Hargrave, R. Platt, R. Savely, and R. Ambrose. "Robonaut 2 - The first humanoid robot in space". In: *2011 IEEE International Conference on Robotics and Automation*. 2011, pp. 2178–2183.

[14] GITAI USA Inc. https://gitai.tech/2021/10/28/iss-tech-demo-ja/.

[15] M. Ekal and R. Ventura. "Disturbance effects observed during micro-gravity experiments on the Astrobee free-flyer and mitigation measures". In: *2024 International Conference on Space Robotics (iSpaRo)*. 2024, pp. 8–14.

[16] D. Hirano, S. Mitani, K. Watanabe, T. Nishishita, T. Yamamoto, and S. P. Yamaguchi. "Int-Ball2: On-Orbit Demonstration of Autonomous Intravehicular Flight and Docking for Image Capturing and Recharging". In: *IEEE Robotics & Automation Magazine* (2024), pp. 2–14.

[17] J. Santos, H. Dinkel, J. Di, P. Borges, M. Moreira, O. Alexandrov, B. Coltin, and T. Smith. "Unsupervised Change Detection for Space Habitats Using 3D Point Clouds". In: *AIAA SCITECH 2024 Forum*. 2024.

[18] NASA. *nasa/astrobee*. https://github.com/nasa/astrobee.

[19] JAXA. *jaxa/int-ball2_simulator*. https://github.com/jaxa/int-ball2_simulator.

[20] N. Koenig and A. Howard. "Design and Use Paradigms for Gazebo, An Open-Source Multi-Robot Simulator". In: *IEEE/RSJ International Conference on Intelligent Robots and Systems*. Sendai, Japan, Sept. 2004, pp. 2149–2154.

[21] Open Robotics. *ISS*. https://app.gazebosim.org/OpenRobotics/fuel/models/ISS.

[22] *moveit/mujoco_ros2_control*. https://github.com/moveit/mujoco_ros2_control.

[23] *ros/xacro*. https://github.com/ros/xacro.

[24] Vention. *Enclosed Ball Screw Actuator, 2295mm Length*. https://vention.io/parts/enclosed-ball-screw-actuator-2295mm-length-1287.

[25] Ewellix. *TLT*. https://www.ewellix.com/en/products/lifting-columns/tlt.

[26] A. Sharp, A. Lovan, E. Herrera, and E. Laske. "Practical End-Effector Development Through Task Interface Taxonomy Analysis". In: *International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*. Vol. 7: 46th Mechanisms and Robotics Conference (MR). Aug. 2022, V007T07A043.

[27] *MoveIt 2*. https://github.com/moveit/moveit2.

[28] I. A. Şucan, M. Moll, and L. E. Kavraki. "The Open Motion Planning Library". In: *IEEE Robotics & Automation Magazine* 19.4 (Dec. 2012), pp. 72–82.

[29] J. van de Weijer, C. Schmid, J. Verbeek, and D. Larlus. "Learning Color Names for Real-World Applications". In: *IEEE Transactions on Image Processing* 18.7 (2009), pp. 1512–1523.

[30] J. Ham, R. Soussan, B. Coltin, H. Chun, and P. Kim. "Drift-Free Visual Compass Leveraging Digital Twins for Cluttered Environments". In: *IEEE Space Mission Challenges for Information Technology / Space Computing Conference Space Robotics Workshop*. 2025.

[31] C. Chamzas, C. Quintero-Pena, Z. Kingston, A. Orthey, D. Rakita, M. Gleicher, M. Toussaint, and L. E. Kavraki. "Motionbenchmaker: A tool to generate and benchmark motion planning datasets". In: *IEEE Robotics and Automation Letters* 7.2 (2021), pp. 882–889.

[32] K. Zakka, Y. Tassa, and MuJoCo Menagerie Contributors. *MuJoCo Menagerie: A collection of high-quality simulation models for MuJoCo*. http://github.com/google-deepmind/mujoco_menagerie. 2022.