2024 International Conference on Space Robotics (iSpaRo)
June 24-27, 2024, Luxembourg

A Modular High-Fidelity Photorealistic Simulator of Orbital Scenarios: A Space Debris Removal Use Case

Xiao LI[1], Antoine Richard[1], Barış C. Yalçın[1], Maxime Hubert Delisle[1],
Miguel Olivares-Mendez[1], and Carol Martinez[1]

2024 International Conference on Space Robotics (iSpaRo) | 979-8-3503-6723-2/24/$31.00 ©2024 IEEE | DOI: 10.1109/iSpaRo60631.2024.10687527

Abstract— **The recent exponential increase in space debris poses a significant and growing threat to the sustainability of space activities. Many Active Debris Removal (ADR) methods and concepts have been proposed over the past decades. However, most of them are still in the technological development stages. Due to the complexity of space missions and the high stakes involved, as well as the complicated and dynamic nature of space environments, the utilization of simulators is paramount in the development of ADR technologies. This article reports the ongoing work on the high-fidelity simulation environment for near-field on-orbit scenarios, which is known as the ALL-STAR Simulator. The orbital dynamics, photo-realistic visualizations, and robotic system implementation being integrated into the simulator are outlined. A specific use case of the ADR mission using the robotic capture system developed by SpaceR is also presented, integrating a gecko-inspired adhesive to the simulator; future work for enabling Software-In-the-Loop and Hardware-In-the-Loop testing for ADR technologies, as well as the extension of diverse space applications, is discussed.**

I. INTRODUCTION

The complexity of space missions and the high risks involved demand substantial financial resources, meticulous preparation, and continuous skill enhancement for engineers and mission control personnel. Therefore, simulators play a crucial role in the space industry by providing a realistic and controlled environment for training, testing, validating, and refining various aspects of space missions. Using simulators enhances safety and efficiency and contributes to cost savings by identifying and addressing potential issues before they arise during missions.

Among space operations, debris removal stands out as one of the most time-sensitive missions due to the proliferation of space debris over the last decades, which represents a significant menace to current operational and future space missions. As the number of non-functional satellites, obsolete rocket parts, and fragments from collisions continues to escalate, the need for active space debris removal has become a pressing concern. To address the space debris challenge, several approaches and technologies have been proposed or are under development [1], such as utilizing harpoon systems [2], net technologies [3], electrodynamic tethers [4], robotic arms [5], and aerodynamic drag augmentation [6]. Nonetheless, due to the complexity of Active Debris Removal (ADR) missions and the prohibitive costs and risks associated with testing innovative technologies in space, most ADR methods are still in their development stages.

The development of a hybrid-compliant system for the soft capture of uncooperative space debris is mentioned in this paper [7]. This robotic capturing system aims to seize debris without damaging it by applying gecko-inspired adhesive material to establish soft and secure contact between the system and debris. This project focuses on the approach phase of ADR missions.

During an ADR mission, precise contact dynamics between the capturing system and debris, accurate orbital motions of the spacecraft, photo-realistic visualization for computer-vision algorithms to intercept and grab debris, and sensor data for control are essential for ensuring the success of the mission. This makes simulators indispensable for ADR missions, particularly during the capturing phase, as they facilitate realistic testing scenarios for precise and delicate manoeuvres required in such operations. Moreover, for the use case in question, that of the robotic capturing system, the simulation of the gecko material [8] is also necessary as it allows for the recreation of the specific context of the ADR mission. Additionally, the controllers of the capturing system must be tested through a Software-In-the-Loop (SIL) simulation, and the system's prototype has to be assessed in a Hardware-In-the-Loop (HIL) robotic facility to simulate microgravity on the ground. Thus, a holistic simulation and testing environment is required, which includes a lifelike depiction of the space environment with microgravity, accurate contact physics and high-resolution visualizations, as well as the ROS (Robot Operating System) interface or alternative platforms such as PyRobot or Webots for SIL and HIL testing. The HELEN project (High-fidELity tEsting enviroNment for active space debris removal), initiated in January 2022, aims to establish this comprehensive simulation environment [9].

Some existing on-orbit scenario simulators cater to various aspects of space exploration, spacecraft operations, trajectory planning and mission design and analysis. For instance, the General Mission Analysis Tool (GMAT) [10] was developed by NASA for general mission analysis purposes, MATLAB/Simulink is capable of simulating satellites in orbit [11], and Blackswan Space [12] is used for satellite mission design. Some of the space simulators focus on space mission engineering and systems analysis, including trajectory planning, modelling, and analyzing space missions and even complex space systems, such as the Ansys Systems Tool Kit (STK) [13]. Certain simulators provide specific

[1]Space Robotics Research Group (SpaceR), SnT Interdisciplinary Centre for Security, Reliability and Trust, University of Luxembourg.
Corresponding author: Xiao LI xiao.li@uni.lu

requirements for explicit space applications, such as the Unreal Rendered Spacecraft On-Orbit (URSO) [14] for Deep learning training for space operations from the University of Surrey, and the Planets And Asteroids Natural Scene Generation Utility (PANGU) [15] developed by the European Space Agency (ESA) is employed in modelling the surfaces of planetary bodies.

The HELEN project simulates, tests, verifies, and validates ADR missions in Low Earth Orbit, focusing on close-range scenarios. To achieve this objective, especially for the use case of the previously mentioned robotic capturing system, there is a list of essential requirements that have to be satisfied.

A. Accurate physics

The accuracy of physics modelling is vital to the success of space debris removal missions, particularly during the critical approach phase. In this stage, when the spacecraft, or robotic system, approaches targeted debris, precise physics simulations are essential to predict and control the complex dynamics occurring in the microgravity environment. Accurate modelling of orbital mechanics, relative velocities, and contact dynamics between the capturing system and debris ensure the faithful simulation of controlled and safe manoeuvres for rendezvous and capture. For ADR missions, sub-millimetre precision is required for near-field orbital trajectory and contact physics during the capturing phase.

B. Robot Operating System Interface

The ADR capture system, a robotic system, is an important use case of the HELEN project. A ROS [16] interface allows users to connect the simulation model of the capture system with the ALL-STAR simulator, facilitating the testing, development, and validation of robotic algorithms in replicated space environments before deploying them on physical prototypes. Thus, it is crucial to integrate a ROS interface into the simulator to test the capture system's controllers using SIL testing and connect the prototype to the on-ground facility for HIL testing.

C. Robot implementation

Given the complexity of the capturing system, comprising joints, actuators, and compliance mechanisms, the simulator must be able to maintain its integrity during implementation. Therefore, every component of the system must be simulated with the expected characteristics, such as geometry and mesh, mass and inertia, coefficients of friction, damping and stiffness, and material properties.

D. Sensor models

Accurate details regarding the positions and rotations of both the debris and the chaser are imperative in active debris removal missions. Hence, incorporating Lidar and Inertial Measurement Units (IMU) sensor models into the simulator is highly beneficial. A camera sensor model can play a vital role in object detection, employing computer vision algorithms to intercept and capture debris. Precise knowledge of the moment of contact during the debris capture phase relies on contact sensors, increasing their significance. Additionally, measuring the force exerted between the debris and the capturing system is crucial at this stage, necessitating the inclusion of a force sensor model in the simulation.

E. Real-time communication

Low-latency Communication between the simulator and the testing facility is of preeminent importance in HIL testing for space systems. This seamless and immediate exchange of information enables the dynamic interactions between the simulated space environment and the physical hardware components being tested. The accuracy of HIL testing relies on the ability of the simulator to emulate realistic scenarios and respond in real-time to inputs from the hardware under examination and the testing facility. This synchronization is crucial for assessing the performance of space systems, validating control algorithms, and testing the integration of hardware components in conditions that mimic the complexities of space.

F. Photo-realistic visualizations

High-resolution visualization in simulators that target ADR missions is significant, particularly for computer vision algorithms. The realistic representation of celestial bodies, spacecraft, and debris enhances the fidelity of simulated scenarios, allowing computer vision algorithms to operate in an environment closely resembling real-world space conditions. For computer vision purposes, accurate and detailed visualizations enable the development and testing of algorithms related to object recognition, navigation, and autonomous decision-making in space.

Table I presents a thorough overview of the state of the art for the most well-known space application simulators on the market. Certain simulators, including GMAT, an open-source software, incorporate commendable orbital physics models that facilitate trajectory computation and planning; commercial software such as Blackswan Space and ANSYS STK offers comprehensive tools and packages capable of meeting the requirements of various space applications. However, a significant impediment lies in terms of commercial accessibility, and according to the authors' knowledge, none of the available simulators can simultaneously satisfy all the aforementioned requirements.

In addition to the extensive requirements of general ADR missions, the robotic capture system necessitates the inclusion of gecko material simulation for on-orbit scenarios. Given the intricate demands inherent to the capture system for ADR missions, the HELEN project endeavours to construct a simulator capable of meeting all specified requirements and addressing current gaps [17] [9]. Furthermore, the envisaged simulator holds the potential for broader applications in diverse scenarios in the future. This simulator is known as the ALL-STAR SIMULATOR from "modulAr high-fideLity photoreaListic near-field teSTing spAce enviRonment".

TABLE I

FEATURE COMPARISON OF POPULAR SPACE SIMULATORS

Software	Physics Accuracy		Interface to other programming languages		Robot implementation	Real-Time	Sensors	Access	Photo-realistic	Parallel Computing
	Orbital	Contact	Software	HIL						
GMAT	Yes	No	Yes	No	No	Modular	Yes	Open-source	Yes	Yes
ANSYS STK	Yes	Yes	Yes	No	Yes	Yes	Yes	Commercial	Yes	Yes
Blackswan Space	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Commercial	Yes	No
MATLAB	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Commercial	No	Yes

TABLE II

FEATURE COMPARISON OF POPULAR SIMULATORS

Software	Physics Engine	Programming languages	Implementation with ROS	Capability of Real-Time communication	Sensor models	Photo-realistic visualizations	Community support	Online resources
Unity	PhysX engine	C#	ROS connector	Yes	Yes	Good overall graphics	Yes	Yes
Unreal Engine	PhysX engine	C++	Unofficial ROS integration code	Yes	Yes	Yes	Yes	Yes
Gazebo	ODE physics engine	C++	ROS connector	Yes	Yes	OpenGL	Yes	Yes
Isaac Sim	PhysX engine	Python environment	ROS bridge	Yes	Yes	GPU rendering	Yes	Yes

II. THE ALL-STAR SIMULATOR

A. Choice of the software

Building the ALL-STAR simulator requires a physics engine that can satisfy all the prerequisites outlined in the preceding section. Therefore, the foremost and most potentially apt physics and game engines were systematically explored, meticulously considering all specified requirements.

1) List of engines: Physics engines [18], such as PhysX (NVIDIA) and Bullet Physics, play a crucial role in emulating interactions between objects in a virtual environment. They handle collisions, friction, gravity, and other physical forces that contribute to creating more realistic simulations. In the context of ADR missions, physics engines must yield precise simulations, ensuring that objects exhibit realistic behaviour in accordance with the principles of physics while accommodating the specifications of the space environment. Furthermore, robotic or game engines serve as comprehensive frameworks for simulation development, encompassing not only physics but also graphics, scripting and additional functionalities. Online resources and community support further facilitate the development process. Table II provides a comparative overview of the commonly utilized and appropriate robotic and game engines for the ALL-STAR simulator, including Unity, Unreal Engine, Gazebo, and Nvidia Omniverse [19].

2) Isaac Sim robotic engine: Following a comprehensive analysis of various potential solutions, ALL-STAR opted for Isaac Sim from Nvidia Omniverse as the robotic simulator for creating on-orbit dynamic scenarios. Isaac Sim provides fast and accurate physics modelling using the PhysX engine, which is critical for simulating the micro-gravity environment and the contact between the ADR system and debris. Thanks to GPU rendering, the software also allows photo-realistic visualizations with controllable render time stepping, which enables users to set the visual parameters according to their specific use case needs.

Moreover, Isaac Sim can be integrated into the HIL system using a ROS bridge and controllable physics time stepping to achieve "real-time" communication. Simulations in Isaac Sim are launched through Python plugins, allowing easy code integration. The Simulator also provides models for cameras, force sensors, contact sensors, IMUs, and Lidar sensors. Overall, Isaac Sim meets all the requirements of the HELEN project and provides additional features that are beneficial for the development of the simulation environment.

B. Concept of operation of the ALL-STAR simulator

Specific on-orbit dynamic scenarios were generated using Isaac Sim, incorporating elements such as a CubeSat featuring the robotic capture system, a model of debris, and a visual representation of Earth. To replicate the microgravity conditions, the gravitational equation needs to be incorporated into Python plugins and subsequently applied for scenario initiation. Additionally, the configuration files must integrate object parameters, further details of which are presented below. The capturing phase stands out as a pivotal aspect in ADR missions, where the precise physics facilitated by Isaac Sim is instrumental in simulating interactions between debris and the ADR system. Furthermore, the mechanical design of the capturing system from third-party software can be imported into the simulations using plugins. Moreover, using the ROS interface, the simulator will be able to connect the virtual world with the HIL on-ground facility. As such, the ADR system will be visualized in the space surroundings while simultaneously undergoing hardware experiments [17].

The ALL-STAR Simulator operates through two primary subsystems, as illustrated in Figure 1: the configuration file and Isaac Sim. The configuration file, a text document, serves as a set of instructions guiding the Simulator in constructing the world/scene. Users can define scenarios within the file, specifying assets such as satellites, ADR systems, debris, and celestial bodies. Initial conditions for on-orbit scenarios and enveloping asset properties, such as position, velocity, and inertial parameters, can be defined by users. Additionally, rendering and contact parameters are configurable in this file.

Through Python, a "world generator" is implemented, utilizing the configuration file as input to produce the world/scene as output. This process involves the construction of an asset library, including planets, various debris types, satellites, and the ADR system. Subsequently, the world generator dynamically assembles the scene by loading the configuration file and assets from the library. The utilization

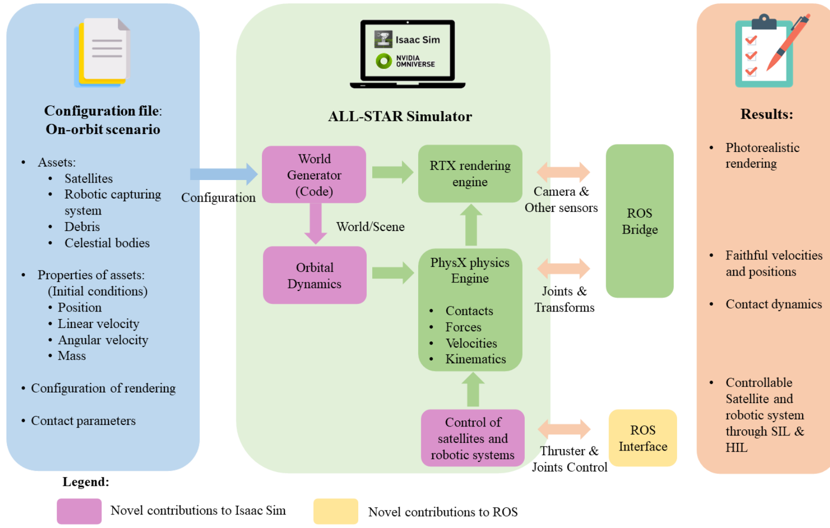

Fig. 1. Concept of operations of the simulator

of Nvidia's RTX rendering engine ensures photorealistic visualization of the scene.

The simulation environment incorporates orbital dynamics by applying gravitational equations through a Python-developed plugin. Upon launching simulations, the PhysX physics engine recreates asset movements in space conditions, encompassing contact dynamics. Furthermore, control algorithms for the ADR system can be integrated into Isaac Sim as a plugin, facilitating SIL and HIL tests. A ROS interface enables connectivity between the ADR system's control algorithms and the testing architectures.

Isaac Sim includes the ROS bridge, facilitating communication with cameras, sensors, and joints. This functionality enables the visualization of the environment and the control of simulated onboard equipment such as joints and thrusters. Consequently, the ALL-STAR Simulator produces simulations characterized by photorealistic rendering, precise states, and accurate contact dynamics. Additionally, it can establish connections with SIL and HIL systems, thereby building a high-fidelity testing framework.

C. Status of the ALL-STAR Simulator

1) Orbital dynamics: The initial phase in constructing the simulation environment involves creating a Python plugin designed for the incorporation of orbital dynamics. The default gravitational effect within Isaac Sim—the average gravitational pull of the Earth along the z-axis—was intentionally disengaged, allowing users to substitute this effect by integrating the equation for universal gravitation.

$$F = G\frac{m_1 m_2}{r^2}$$

In this manner, fundamental interactions among celestial objects can be simulated within the framework of Isaac Sim. Basic orbital movements were verified through simulation tests, with one scenario replicating the solar system created within Isaac Sim's interface. The simulation featured elemental shapes representing the eight planets, and real-life scaled values for mass, initial orbital velocity, and initial positions were integrated into the script. As a result, the planetary dimensions, inter-object distances, and velocities adhere to realistic proportions on a real-world scale.

Following the successful integration of orbital physics into the simulator, the subsequent phase involved recreating on-orbit scenarios tailored for ADR missions within Low Earth Orbits. A primary depiction of a satellite circling the visual Earth was reconstructed inside Isaac Sim. The resulting simulation data was systematically documented in CSV files and subsequently compared with the outcomes derived from analytical calculations corresponding to identical parameters.

However, during the comparative analysis of simulation and analytical results, an inherent limitation in precision within the PhysX physics engine surfaced. The physics engine applies the single-precision floating-point format (float32), implying that in celestial-scale long-range simulations, the precision of the trajectories is constrained to approximately 0.5 meters per step. Therefore, for users requiring contact dynamics emulation between objects while simulating far-field orbital trajectories, the available precision may prove insufficient for their specific requirements. Nonetheless, in the context of the ALL-STAR Simulator's targeted applications in close-range space environments (within a distance of several kilometres), it is possible to discount or circumvent this limitation.

2) Photo-realistic visualisations: A central component within scenarios involving satellite pursuit of targeted debris in Low Earth Orbit frequently entails the utilization of a virtual depiction of Earth. To this end, a visual model of Earth was delicately developed within the Isaac Sim interface, as shown in Figure 2, employing basic sphere geometries scaled to mirror real-world proportions. Authentic Earth imagery sourced from NASA Visible Earth [20] was incorporated to delineate the model's surface. Furthermore, supplementary layers, such as clouds and atmospheric elements, were integrated into the Earth model's surfaces to enhance its fidelity and realism.

Fig. 2. Virtual depiction of the Earth (NASA visible earth)

The inclusion of debris representation is imperative in simulating ADR missions, with high-resolution assets proving advantageous for computer vision algorithms. Several open-source repositories offer high-quality imagery readily accessible through online resources, enabling its integration into the on-orbit scenario with necessary modifications tailored to meet specific requirements.

Throughout ADR missions, a camera within the Isaac Sim interface may be configured to track the movements of the chaser satellite closely, facilitating comprehensive documentation of the approaching and capturing phases.

3) Implementation of the robotics systems: Various types of robotic files, such as the Unified Robot Description Format (URDF) and the MuJoCo Modeling XML File (MJCF), can be imported into Isaac Sim, enabling users to incorporate specific designs into simulations. Furthermore, the Onshape plugin expands the importation capabilities to include Computer-aided design (CAD) files, thereby accommodating a broader range of mechanical designs.

4) Near-field on-orbit simulation: The ALL-STAR simulator is centred on close-range on-orbit simulations, which play a critical role in recreating any ADR mission. The absence of specialized simulators suited for near-field operations highlighted the necessity of developing the ALL-STAR Simulator. Within the scope of ALL-STAR's targeted applications, the distance between debris and the robotic system under scrutiny is deliberately confined to several kilometres. Consequently, the influence of specific inherent space environmental conditions, such as magnetic fields and solar radiation, can be mitigated to a certain extent. Moreover, the precise determination of the relative distance and velocity between spacecraft within an orbital environment is of paramount importance for executing close-range operations, particularly in the context of ADR missions during approach and capture stages. Given the confinement of relative distances to limited space, the accuracy afforded by the physics engine PhysX proves adequate for simulating such missions effectively.

III. Space Debris Removal Use Case

A. Virtual Earth and Debris model

The preceding section details the simulator's underlying attributes; several other characteristics were then incorporated to tailor the simulator to the ADR and HELEN contexts. For the specific use case of the HELEN project, in addition to the robotic capturing system, fundamental orbital scenarios must encompass a targeted debris model and a virtual Earth model.

B. The capturing system model

For the HELEN use case, a crucial step in creating the simulations involves the integration of the capturing system into the simulator framework. Originally designed as CAD files within Autodesk Fusion 360, this system requires conversion into URDF prior to importation into Isaac Sim. The primary challenge arises during importation, as all joints and actuators and their functionality must be preserved. However, the conversion process mandates the initial exportation of the CAD file into a Xacro file, utilizing an unofficial Fusion 360 plugin, followed by subsequent conversion into a URDF file. Due to the unofficial nature of the plugin, shortcomings emerged, particularly in the proper importation of joints and axes within Isaac Sim. During the process of implementation iteration, an alternative integration approach was explored utilizing Isaac Sim's Onshape plugin.

The design of the capturing system was transferred into Onshape and then imported into Isaac Sim using the interface's dedicated plugin. This method simplifies the integration procedure and provides more reliable results. With appropriate adjustments, the imported joints and actuators of the capturing system were found to meet the anticipated criteria.

Upon successful integration of the ADR system into the simulator, the following phases, including the final approach towards the debris, the capture, and post-capture operations, can be executed and evaluated as in Figure 3.

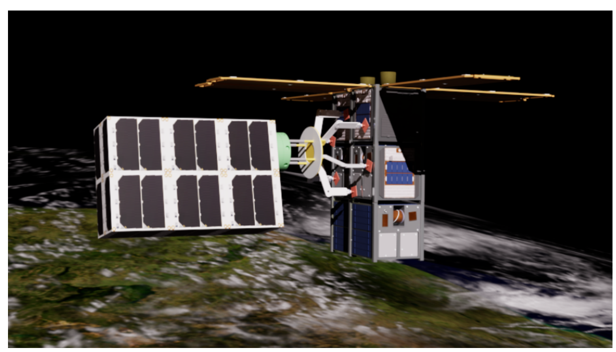

Fig. 3. Simulation of the approach phase of an ADR mission

C. Applying force in a complex robotic system

Upon the launch of the ADR mission simulations for the capturing system, an initial force and velocity were applied to simulate real-world conditions within the chosen Low Earth Orbit parameters. Given the composite nature of the capturing system, comprising numerous rigid bodies and joints with varying masses and shapes, the distribution of mass is non-uniform and subject to variation over time. Thus, for the precise application of initial values, the actual centre of mass for the entire system must be precisely calculated at every step of the simulations, accounting for each component.

Another challenge is USD (Universal Scene Description), the underlying technology utilized by Isaac Sim, developed by Pixar. Within the USD, an "Xform" (an abbreviation of "transform") serves as a fundamental concept, representing a transformation matrix within the context of USD Prim (short for "primitives"). The Xform facilitates control over the position, rotation, and scale of objects within a scene. Consequently, both the capturing system and the debris model were allocated to Xforms. However, a complication arose when applying the initial force at the system's center of mass within its Xform frame, as this led to the robotic system spinning unintentionally during simulation. Because the capturing system is an articulated robot with multiple joints, the Xform's frame does not translate with the object when force is applied. In order to resolve this, the force was applied to the local frame of the robot's base link at coordinates adjusted for the center of mass within the frame.

D. Material simulation

There are multiple requirements for the material used in the capturing system. Firstly, a "soft" approach is desired, in which the debris is not harmed during capture. Secondly, the material must adapt to different surfaces, and finally, ideally, it would permit multiple attempts at capture. The material that was chosen to offer all of these characteristics was gecko material [8].

The robotic capture system employs bio-inspired micro-patterned dry adhesives that mimic the unique adhesive properties observed on geckos' feet. Geckos have the remarkable ability to climb vertical surfaces and adhere to ceilings due to the intricate hierarchical structures on their footpads, which consist of millions of tiny hair-like structures called setae. These setae branch out into even smaller structures called spatulae, which create a large surface area for molecular interactions known as van der Waals forces [21].

Researchers have developed synthetic adhesives inspired by the structure and function of gecko feet [22]. These adhesives typically consist of arrays of microscopic pillars or fibres, mimicking the structure of setae and spatulae. The high surface area and flexibility of these structures allow them to conform to rough surfaces and create strong adhesion through van der Waals forces.

These adhesives typically demonstrate excellent repeatability and re-usability over numerous attachment and detachment cycles. They can adhere to various surfaces, including smooth, rough, dry, and even dirty surfaces, and maintain their adhesive properties throughout multiple cycles of attachment and detachment.

The adaptability to various surfaces and the re-usability of gecko-inspired adhesives make them exceptionally appealing for ADR applications, wherein the capability to grasp and release debris repeatedly without inflicting damage is greatly valued [23] [24].

1) The difficulty of gecko material simulation: Simulating gecko-inspired adhesives, however, presents several challenges due to the complex nature of both the biological adhesive system and its synthetic counterparts. Replicating the hierarchical structure of gecko foot pads requires capturing interactions at multiple length scales, from nanometers to micrometres, which can be computationally intensive and technically challenging. Multi-scale modelling [25] techniques integrate simulations at different lengths and time scales to capture the hierarchical structure and behaviour of gecko-inspired adhesives across multiple levels of complexity; however, they also present elevated difficulty. Finite Element Modelling (FEM) [26] is commonly used to emulate the mechanical behaviour of these adhesive structures, which requires significant computational resources and expertise. In addition, modelling the inter-molecular forces between the adhesive and targeted surfaces with high precision demands sophisticated Molecular Dynamics (MD) simulations [27].

2) Gecko material simulation for the HELEN use case: The robotic capturing system incorporates a gecko-inspired adhesive material developed by external institutions. Consequently, the fundamental functional characteristics of this material, alongside its integration within the space environment simulator, have piqued the interest of the HELEN project. Rather than employing molecular structure simulations to explore the gecko adhesive material at the microscopic level utilizing methodologies discussed earlier, simulations focusing on macroscopic and functional behaviours suffice and align well with the requirements of the capturing system.

The gecko adhesive materials are integrated as a thin layer under the pads at the tip of each of the legs of the capturing system [7]. During the capture phase of an ADR mission, these pads, coupled with the dry adhesive material, will be brought into contact with one of the surfaces of the targeted debris and, once all the conditions for the activation of the material are met, will stick to the surface without causing any harm.

During an ADR mission, the simulation of this adhesive material's functional characteristics within the framework of the robotic capturing system can be outlined in the following sequence of steps. Initially, the distance between the gecko material beneath each pad and the debris surface is maintained below a predefined threshold. Subsequently, alignment is ensured between the pads and the targeted debris surface, with the coplanarity of both surfaces confirmed through the calculation of cross-products of their normal vectors. Thirdly, contact is established between the surfaces within the same spatial domain, requiring the implementation of a basic algorithm to verify surface overlap. Fourthly, the total contact area surpasses a specified minimum threshold, and contact duration is guaranteed. The determination of the intersection polygon between parallel surfaces utilizes the Sutherland–Hodgman Algorithm [28] in simulations, facilitating the calculation of the contact area based on the algorithm's outcomes. Finally, activation of the gecko material necessitates the application of a shear force to the pads, which can be achieved by using Isaac Sim's intricate functions.

Additional tests of the gecko-inspired adhesive materials are currently being conducted in the zero-G robotic facility of the University of Luxembourg. The parameters of the materials employed in the simulations have been adjusted to replicate those documented in the work of the external institutes [29], along with the corresponding test outcomes. Therefore, it is anticipated that the essential attributes of the gecko-inspired materials in simulations will align with the actual specifications at a macroscopic level. Lower computational demands and greater suitability for the particular application scenario characterize the simulation approach applied by the HELEN project.

IV. Future Plans

Alongside its current functionalities, the ALL-STAR simulator is anticipated to incorporate SIL and HIL testing systems to test, verify and validate the capturing system. Additionally, the scope of the simulator's applications can be expanded beyond ADR missions to include other use cases.

A. Robot Operating System interface

A foreseen development within the simulator is the creation of a ROS interface, which will facilitate controller testing for both the capturing system and its associated satellite. This interface will incorporate sensor models, enabling users to manipulate the actuators and joints of the capturing system while also receiving sensor data. Furthermore, integration of the interface with the robotic facility will enable real-time connectivity to the HIL testing system.

B. Software-In-the-Loop testing for the controllers

The SIL testing system will be utilized to assess the decision algorithms and capture controllers of the capturing system via the ROS interface. The simulator integrates various aspects of the space environment, encompassing orbital dynamics and scenes. Within this framework, models representing the capturing system, the associated CubeSat, debris, and their associated parameters and states are incorporated. Throughout the simulations, the orbital states of the systems will be simulated and conveyed to the sensor models. For instance, the camera system is capable of detecting debris, determining its position, and identifying fundamental characteristics. Meanwhile, the IMU sensor can measure the CubeSat's linear acceleration, angular rate, and rotational dynamics, providing inputs to the decision algorithms. These algorithms, which can be integrated into Isaac Sim via a Python plugin, process the sensor measurements to generate commands. Subsequently, these commands will be transmitted to the actuator models of the capturing system. The CubeSat and the mechanic system models, in turn, receive these commands through the ROS interface, adjusting their movements accordingly. Throughout the interaction between the debris and the robotic system, a force sensor will be utilized to measure the contact force.

C. Hardware-In-the-Loop testing with the Zero-G robotic facility

The zero-gravity robotic facility utilizes UR10 (Universal Robots) robots mounted on rail systems to replicate space scenarios under microgravity conditions. Positioned at the end effector of these robotic arms are the mockup of debris and the prototype of the capturing system, configured to mimic orbital movements in a Low Earth Orbit environment. The UR10 arms are operated by the facility's control system, which will be linked to the simulator to enable real-time communication with the HIL testing system. Consequently, during hardware tests, the simulator must be capable of commanding the robotic arms and promptly receiving measured forces and moments acting on the prototype's base. It is anticipated that the robotic arms will responsively execute commands from the simulator and spontaneously transmit sensor data to it.

D. Expand the use case for other near-field applications

Beyond ADR missions, it is envisaged that the ALL-STAR simulator will have the capability to support a range of space mission simulations, enabling users to replicate diverse close-range scenarios. The simulator is expected to adopt a modular approach to accommodate other near-field space activities, including docking manoeuvres, robotic manipulation in space, and close-range Guidance, Navigation, and Control (GNC) tests. In meeting control requirements, the simulator will incorporate computer vision functionalities through the integration of OpenCV. Additionally, given the need to expand the range of scenarios for various applications, the incorporation of a synthetic data generator to create scenarios will be beneficial.

V. Conclusion

Given the complex and demanding nature of space missions, the utilization of simulators plays a crucial role in the development of space technologies. Specifically, for the approach and capture phases of ADR missions, numerous crucial prerequisites, ranging from precise physics simulations to photo-realistic visualizations, underscore the need for a simulator capable of replicating a realistic testing environment. The absence of specialized space simulators for close-range scenarios served as motivation for the HELEN project to undertake the development of its simulator. This initiative aims to facilitate the testing, verification, and validation of the robotic capturing system for ADR missions.

The HELEN project utilizes the Isaac Sim robotic simulation toolkit within the Nvidia Omniverse to construct the ALL-STAR Simulator, a high-fidelity simulation environment tailored for near-field on-orbit scenarios. This simulator integrates precise physics, high-resolution imagery, and the implementation of robotic systems designed for ADR missions. Specifically accommodated to the robotic capturing system, the simulation incorporates the core functional characteristics and behaviour of gecko-inspired dry adhesive. This article also presents the envisioned development of a ROS interface to facilitate SIL and HIL testing for the capturing system's controllers and hardware prototype and potential future expansions of the ALL-STAR Simulator's applications in space. Overall, the simulator is well-poised to contribute to the testing of ADR technologies and other space missions within near-field scenarios.

ACKNOWLEDGMENT

This study has been partially funded by Luxembourg National Research Fund (FNR) - BRIDGES funding for "High fidELity tEsting eNvironment for Active Space Debris Removal - HELEN", project ref: BRIDGES2021/MS/15836393.

REFERENCES

[1] B. C. Yalçın, C. Martinez, M. Hubert Delisle, G. Rodriguez, J. Zheng, and M. Olivares-Mendez, "ET-Class: An Energy Transfer-Based Classification of Space Debris Removal Methods and Missions," *Frontiers in Space Technologies*, vol. 3, p. 792944, Mar. 2022. [Online]. Available: https://www.frontiersin.org/articles/10.3389/frspt.2022.792944/full

[2] R. Dudziak, S. Tuttle, and S. Barraclough, "Harpoon technology development for the active removal of space debris," *Advances in Space Research*, vol. 56, no. 3, pp. 509–527, Aug. 2015. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S0273117715002719

[3] M. Shan, J. Guo, and E. Gill, "Deployment dynamics of tethered-net for space debris removal," *Acta Astronautica*, vol. 132, pp. 293–302, Mar. 2017. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S0094576516309961

[4] V. I. Trushlyakov and V. V. Yudintsev, "Rotating tethered system for active space debris removal," *Journal of Physics: Conference Series*, vol. 1260, no. 11, p. 112032, Aug. 2019. [Online]. Available: https://iopscience.iop.org/article/10.1088/1742-6596/1260/11/112032

[5] W. Zhang, F. Li, J. Li, and Q. Cheng, "Review of On-Orbit Robotic Arm Active Debris Capture Removal Methods," *Aerospace*, vol. 10, no. 1, p. 13, Dec. 2022. [Online]. Available: https://www.mdpi.com/2226-4310/10/1/13

[6] L. Visagie, V. Lappas, and S. Erb, "Drag sails for space debris mitigation," *Acta Astronautica*, vol. 109, pp. 65–75, Apr. 2015. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S0094576514005190

[7] M. Hubert Delisle, O.-O. Christidi-Loumpasefski, B. C. Yalçın, X. Li, M. Olivares-Mendez, and C. Martinez, "Hybrid-Compliant System for Soft Capture of Uncooperative Space Debris," *Applied Sciences*, vol. 13, no. 13, p. 7968, Jul. 2023. [Online]. Available: https://www.mdpi.com/2076-3417/13/13/7968

[8] M. K. Ben-Larbi, R. Hensel, G. Atzeni, E. Arzt, and E. Stoll, "Orbital debris removal using micropatterned dry adhesives: Review and recent advances," *Progress in Aerospace Sciences*, vol. 134, p. 100850, Oct. 2022. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S0376042122000422

[9] X. Li, A. Richard, O. Christidi-Loumpasefski, B. C. Yalcin, C. Martinez, M. Hubert, G. Rodriguez, J. Zheng, and M. O. Mendez, "Nvidia Omniverse for Active Space Debris Removal Missions, an Overview."

[10] S. P. Hughes, R. H. Qureshi, S. D. Cooley, and J. J. Parker, "Verification and Validation of the General Mission Analysis Tool (GMAT)," in *AIAA/AAS Astrodynamics Specialist Conference*. San Diego, CA: American Institute of Aeronautics and Astronautics, Aug. 2014. [Online]. Available: https://arc.aiaa.org/doi/10.2514/6.2014-4151

[11] A. Shirazi, J. Ceberio, and J. A. Lozano, "Simulation Framework for Orbit Propagation and Space Trajectory Visualization," *IEEE Aerospace and Electronic Systems Magazine*, vol. 36, no. 8, pp. 4–20, Aug. 2021. [Online]. Available: https://ieeexplore.ieee.org/document/9508476/

[12] V. Muralidharan, C. Martinez, A. Zinys, M. Klimavicius, and M. Olivares-Mendez, "Autonomous control for satellite rendezvous in near-Earth orbits," in *2022 International Conference on Control, Automation and Diagnosis (ICCAD)*. Lisbon, Portugal: IEEE, Jul. 2022, pp. 1–6. [Online]. Available: https://ieeexplore.ieee.org/document/9853882/

[13] J. Carrico, D. Cather, G. Corrao, M. Exnicious, L. Kay-Bunnell, N. F. K. Policht, N. Kinzly, R. Lee, J. Locke, M. Loucks, Z. Moghal, N. Popel, J. Ramrath, C. Short, and N. Stankey, "REVISITING TRAJECTORY DESIGN WITH STK ASTROGATOR PART 3."

[14] P. F. Proenca and Y. Gao, "Deep Learning for Spacecraft Pose Estimation from Photorealistic Rendering," Aug. 2019, arXiv:1907.04298 [cs]. [Online]. Available: http://arxiv.org/abs/1907.04298

[15] N. Rowell, S. Parkes, M. Dunstan, and O. Dubois-Matra, "PANGU: VIRTUAL SPACECRAFT IMAGE GENERATION."

[16] J. M. O'Kane, *A gentle introduction to ROS*, version 2.1.1 (3e3d9c5), generated in november 20, 2014 ed. Columbia, SC: Jason M. O'Kane, 2014.

[17] X. Li, B. C. Yalcin, C. Martinez, O. Christidi-Loumpasefski, M. H. Delisle, G. Rodriguez, J. Zheng, and M. Olivares-Mendez, "Exploring NVIDIA Omniverse for Future Space Resources Missions."

[18] T. Erez, Y. Tassa, and E. Todorov, "Simulation tools for model-based robotics: Comparison of Bullet, Havok, MuJoCo, ODE and PhysX," in *2015 IEEE International Conference on Robotics and Automation (ICRA)*. Seattle, WA, USA: IEEE, May 2015, pp. 4397–4404. [Online]. Available: http://ieeexplore.ieee.org/document/7139807/

[19] J. Collins, S. Chand, A. Vanderkop, and D. Howard, "A Review of Physics Simulators for Robotic Applications," *IEEE Access*, vol. 9, pp. 51 416–51 431, 2021. [Online]. Available: https://ieeexplore.ieee.org/document/9386154/

[20] NASA, "NASA Visible Earth, Blue Marble Next Generation," Dec. 2004, publisher: NASA Visible Earth. [Online]. Available: https://visibleearth.nasa.gov/images/74218/december-blue-marble-next-generation/74220l

[21] M. Zhou, N. Pesika, H. Zeng, Y. Tian, and J. Israelachvili, "Recent advances in gecko adhesion and friction mechanisms and development of gecko-inspired dry adhesive surfaces," *Friction*, vol. 1, no. 2, pp. 114–129, Jun. 2013. [Online]. Available: https://link.springer.com/10.1007/s40544-013-0011-5

[22] L. Wang, Y. Hui, C. Fu, Z. Wang, M. Zhang, and T. Zhang, "Recent advances in *Gecko* -inspired adhesive materials and application," *Journal of Adhesion Science and Technology*, vol. 34, no. 21, pp. 2275–2291, Nov. 2020. [Online]. Available: https://www.tandfonline.com/doi/full/10.1080/01694243.2020.1760478

[23] M. A. Estrada, B. Hockman, A. Bylard, E. W. Hawkes, M. R. Cutkosky, and M. Pavone, "Free-flyer acquisition of spinning objects with gecko-inspired adhesives," in *2016 IEEE International Conference on Robotics and Automation (ICRA)*. Stockholm, Sweden: IEEE, May 2016, pp. 4907–4913. [Online]. Available: http://ieeexplore.ieee.org/document/7487696/

[24] H. Jiang, E. W. Hawkes, C. Fuller, M. A. Estrada, S. A. Suresh, N. Abcouwer, A. K. Han, S. Wang, C. J. Ploch, A. Parness, and M. R. Cutkosky, "A robotic device using gecko-inspired adhesives can grasp and manipulate large objects in microgravity," *Science Robotics*, vol. 2, no. 7, p. eaan4545, Jun. 2017. [Online]. Available: https://www.science.org/doi/10.1126/scirobotics.aan4545

[25] T. Materzok, H. Eslami, S. N. Gorb, and F. Müller-Plathe, "Understanding Humidity-Enhanced Adhesion of Geckos: Deep Neural Network-Assisted Multi-Scale Molecular Modeling," *Small*, vol. 19, no. 22, p. 2206085, Jun. 2023. [Online]. Available: https://onlinelibrary.wiley.com/doi/10.1002/smll.202206085

[26] J. Olender, J. Perris, Y. Xu, C. Young, D. M. Mulvihill, and N. Gadegaard, "Gecko-inspired dry adhesives for heritage conservation – tackling the surface roughness with empirical testing and finite element modelling," *Journal of Adhesion Science and Technology*, vol. 37, no. 6, pp. 1091–1116, Mar. 2023. [Online]. Available: https://www.tandfonline.com/doi/full/10.1080/01694243.2022.2061153

[27] T. Materzok, S. Gorb, and F. Müller-Plathe, "Gecko adhesion: a molecular-simulation perspective on the effect of humidity," *Soft Matter*, vol. 18, no. 6, pp. 1247–1263, 2022. [Online]. Available: http://xlink.rsc.org/?DOI=D1SM01232K

[28] A. Rappoport, "An efficient algorithm for line and polygon clipping," *The Visual Computer*, vol. 7, no. 1, pp. 19–28, Jan. 1991. [Online]. Available: http://link.springer.com/10.1007/BF01994114

[29] E. W. Hawkes, H. Jiang, and M. R. Cutkosky, "Three-dimensional dynamic surface grasping with dry adhesion," *The International Journal of Robotics Research*, vol. 35, no. 8, pp. 943–958, Jul. 2016. [Online]. Available: http://journals.sagepub.com/doi/10.1177/0278364915584645