UKAEA-RACE-PR(22)06

Alice Cryer, Alfie Sargent, Fumiaki Abe, Paul Dominick Baniqued, Ipek Caliskanelli, Hasan Kivrak, Hanlin Niu, Salvador Pacheco-Gutierrez, Alexandros Plianos, Masaki Sakamoto, Tomoki Sakaue, Wataru Sato, Shu Shirai, Yoshimasa Sugawara, Harun Tugal, Myles Verdon, Andika Yudha, Robert Skilton

Digital Mock-Ups for Nuclear Decommissioning: A survey on existing simulation tools for industry applications

Digital Mock-Ups for Nuclear Decommissioning: A survey on existing simulation tools for industry applications

Alice Cryer, Alfie Sargent, Fumiaki Abe, Paul Dominick Baniqued, Ipek Caliskanelli, Hasan Kivrak, Hanlin Niu, Salvador Pacheco-Gutierrez, Alexandros Plianos, Masaki Sakamoto, Tomoki Sakaue, Wataru Sato, Shu Shirai, Yoshimasa Sugawara, Harun Tugal, Myles Verdon, Andika Yudha, Robert Skilton

This is a preprint of a paper submitted for publication in
Frontiers in Robotics and AI

Digital Mock-Ups for Nuclear Decommissioning: A survey on existing simulation tools for industry applications

Alice Cryer [1,*], **Alfie Sargent** [1], **Fumiaki Abe** [2], **Paul Dominick Baniqued** [3], **Ipek Caliskanelli** [1], **Hasan Kivrak** [3], **Hanlin Niu** [3], **Salvador Pacheco-Gutierrez** [1], **Alexandros Plianos** [1], **Masaki Sakamoto** [2], **Tomoki Sakaue** [2], **Wataru Sato** [2], **Shu Shirai** [2], **Yoshimasa Sugawara** [2], **Harun Tugal** [1], **Andika Yudha** [1] **and Robert Skilton** [1]

[1] *Remote Applications in Challenging Environments (RACE), UK Atomic Energy Authority, Culham Science Centre, Abingdon, Oxfordshire, United Kingdom*
[2] *TEPCO, 1-1-3 Uchisaiwai-cho, Chiyoda-ku, Tokyo, Japan*
[3] *University of Manchester, Oxford Rd, Manchester, United Kingdom*

Correspondence*:
Alice Cryer
alice.cryer@ukaea.uk

ABSTRACT

The maturation of Virtual Reality software introduces new avenues of nuclear decommissioning research. Digital Mockups are an emerging technology which provide a virtual representation of the environment, objects or processes, supporting the whole lifecycle of product development and operations. This paper provides a survey on currently available simulation tools to design digital mock-ups required for safe remote decommissioning activities in the nuclear industry. The survey looks at eleven simulation tools; CoppeliaSim, Gazebo, Ignition, Nvidia Omniverse Isaac Sim, WeBots, Choreonoid, AGX Dynamics, MORSE, VR4Robots, RoboDK, and Toia. Using the available documentation, the different capabilities of these software packages were assessed for their suitability to nuclear decommissioning; such as environment simulation, haptic interfaces, and general usability.

1 INTRODUCTION

Accomplishing safe and effective nuclear decommissioning is an ongoing global challenge. The ALARA (as low as reasonably achievable) principle is a key concept in intervention planning, requiring constant research into new techniques to reduce occupational exposure to radiation. One major technique is the deployment of robotic solutions into the decommissioning environment instead of a human worker. This is also called *remote handling*, and is a cornerstone of the modern nuclear decommissioning process. However this approach introduces new challenges that must be taken in consideration when designing a suitable remote handling system:

- A significant amount of background radiation must be anticipated, and robotic manipulators could be in direct contact with radioactive sources. This means that robotic components must be either made to be radiation tolerant, and/or be easily replaceable in case of failure. If the latter, considerations must be made for how this maintenance would be carried out, as human presence in high radiation areas is limited.
- The environment of decommissioning sites is often unstructured, for example the Fukushima Daiichi Power Plant after the nuclear disaster. This requires robotic systems to include sensors to map their surroundings, and capable of navigating around obstacles and manoeuvring in tight spaces.

The recent rise and maturation of virtual reality (VR) and simulation software has lead to the research and development of new tools, such as Digital Mock-Ups (DMUs) and Digital Twins. A Digital Twin (DT) is the digital representation of a physical environment, machine, or structure, whose state that is (at least) periodically updated to reflect the physical object's actual state. DMUs are interactive digital models that are used for mock-up purposes - such as training, design, testing, etc. DMUs are distinct from DTs: DTs are used to mirror the physical and the virtual, while DMUs are interactive and do not necessarily reflect the current physical state of what they represent.

Virtual Reality is a computer-generated visualisation technique where users both experience and interact with an artificial three dimensional audio-visual (and sometimes tactile) environment. A simulation is a model of a system or process, and is used to assess defining parameters and mechanisms, and can also be used to predict future behaviour. Simulations are not interactive: VR software simulates an interactive virtual environment, however while the parameters of the simulation can be altered, the underlying process of "simulation" cannot.

DMUs open a new avenue of remote handling research: the development of a DMU that brings together virtual reality and simulation software with live robotic sensor data. The use of a DMU would give operators more information on the state of the environment, presenting several advantages for both planning and remote maintenance operations. The aims of the DMU would be:

- Accelerate strategy development
- Assist in identifying and developing operator skills required for the remote maintenance tasks
- Provide a test bed to design, optimise, and test: tools, equipment, and operations, prior to robot deployment
- Provide live-stream data to augment operators' understanding of the decommissioning environment and remote maintenance tasks during the deployment
- Collate and review deployments in order to learn, feeding back into the strategy development aspect (point 1)

This paper is a survey of existing simulation and VR software, using their documentation to assess the features each of them provide. The focus is on their potential to create a Next Generation Digital Mock-up (NG-DMU), a concept for a future DMU with enhanced function, interoperability, and performance, for a nuclear decommissioning use-case. This paper will include a literature review of the relevant areas of interest for a nuclear decommissioning NG-DMU. The review will look at current research and deployment in the nuclear industry of: simulation, virtual reality, and deep learning tools; the use of robotics, and the use of digital twins. The key software features of the simulation tools have been identified, with overview of how these relate to the above aims for the creation of a nuclear decommissioning NG-DMU. The simulation tools will be reviewed, presenting their main user-base and the prominent features of the software. This overview will lead into and inform a comparison between the tools; how their features compare and contrast, and potentially used together in a complementary setup.

2 LITERATURE REVIEW

This section will review the existing literature regarding the relevant areas of interest for a nuclear decommissioning NG-DMU. The review has been divided into three categories: the current research and deployment of simulation and virtual reality tools in the nuclear industry; the use of robotics and the use of digital twins in the nuclear industry, and how deep learning can be used in DMUs and in the nuclear industry.

2.1 Simulation and Virtual Reality tools for the Nuclear Industry

Simulation is a powerful technique to predict the future state of the simulated object, or environment. It is particularly interesting for radioactive environments as it allows for the environment to be investigated without the need for physical presence - removing the risk to personnel and electronic systems. Conventional radiation simulations using the Monte Carlo method are computationally intense, and time consuming. A new technique for gamma dose estimations using the point kernel method and CAD was developed as a more efficient and flexible alternative, by Liu et al. (2022), even allowing the simulation environment to be updated online. The accuracy was verified against MCNP, and found to be reliable within the set parameters: 01-10 MeV photons and 0-20 mfp shielding thickness. Simulations can also provide insight into proposed decommissioning methodologies, such as cutting in work by Williams et al. (2011), and Hyun et al. (2017). Nash et al. (2018) successfully integrated VR hand controllers in training decommissioning simulations, where HTC Vive hand controllers were used to provide input and haptic feedback to a remote teleoperation task.

Immersive virtual reality applications can be used in a range of applications in the nuclear industry, such as visualizing and assessing different maintenance procedures like refuelling as in work by Jin-Yang et al. (2020), or for training, where Cryer et al. (2019) developed a platform where virtual dosimeters can track worker doses during a decommissioning training scenario. The maturation of virtual reality has lead to its development for future applications in nuclear environments, including nuclear fusion, such as work by Gazzotti et al. (2021).

2.2 Robotics and Digital Twins in the Nuclear Industry

The deployment of robots (then known as remote systems) in the nuclear industry has been implemented since the 1940s, and is relatively as old as nuclear research itself, (Wehe et al., 1989). Such systems were mainly developed to protect human operators from hazardous environments during typical scenarios but have since expanded their application to decommissioning and surveillance of serious safety incidents.

Early robots have played a critical part in the remote inspection and recovery operations in major nuclear disasters such as Chornobyl and the Three Mile Island incidents (Wehe et al., 1989), (Adamov and Yegorov, 1987), (Gelhaus and Roman, 1990). In nuclear decommissioning, most of the tasks developed for robots are related to inspection and handling. Remote inspection involves using robot sensors (i.e. vision, geometric, environmental) to scan the facility and gather data for future use. For example, (Groves et al., 2021) have shown a mobile inspection robot can use its LIDAR (Light Detection and Ranging) system, cameras, and radiation detectors to explore and map an unknown nuclear facility environment while avoiding hot spots of ionising radiation. The generated map can then be used to plan future missions where more active tasks such as remote handling and teleoperation are involved.

(Connor et al., 2020) have successfully mapped 15 km^2 around the Chornobyl nuclear power plant using a fixed-wing unmanned aerial system (UAS). This demonstrated that UASs can be deployed on radiation mapping surveys and return to safe areas afterwards. In addition to the other findings, a localized hot-spot previously unreported in literature was discovered in the survey area using the UAS.

Risk-aware robotics have also been researched in the sense of inspection, as seen in Barbosa et al. (2021): a risk-cost function can be used to calculate the path of minimal cost, when the robot's motion-planning algorithm has no prior model of the environmental hazards. The function was demonstrated using both sampling-based and optimisation-based approaches, where the robot's goal was to move from the initial state to the target area, without modeling the hazard beyond the samples taken en-route.

Surveys such as these require novel radiation detectors which are low-cost and easy to deploy. Verbelen et al. (2021) developed a miniaturised gamma-scanning platform for decommissioning scenarios, the 'CC-RIAS', for the purpose of environment mapping and radioactive waste characterization. The system was specifically designed to be small enough to deploy through access ports in nuclear sites, and includes a commercial CZT gamma spectrometer and a motorised pan-tilt base.

Radiation-hardened or radiation-tolerant electronics is also important to research, in particular power systems which are sensitive to radiation. (Verbelen et al., 2022) integrated a buck-boost converter circuit into a radiation inspection instrument and then deployed it at the Chornobyl Nuclear Power Plant. It was exposed to an integrated dose in excess of 0.3 mGy over 2 weeks of field work, with no failures observed.

Nuclear robots can also be categorised based on the environment or scenario where they are operated. For example, ground robots which come in various forms (i.e. legged, wheeled, tracked, etc.) can be used to survey human-level operations. They are deployed based on their mechanical capabilities of moving through terrain: wheeled robots are limited to relatively smooth surfaces free of clutter, seen in Groves et al. (2021), whereas legged robots such as quadrupeds and hexapods can move through obstacles and navigate through stairwells, shown by (Wisth et al., 2019), and (Cheah et al., 2019). On the other hand, aquatic robots designed to traverse the surface (e.g. MallARD Groves et al. (2019)) or be submerged underwater (e.g. AVEXIS Nancekievill et al. (2018) and BlueROV2 Blue Robotics (2022)) can be deployed in water tanks and other reservoirs, while drones or Unmanned Aerial Vehicles (UAV) are used primarily to scan and capture images of the environment in areas where ground or aquatic robots are not able to reach. Remote handling operations involve the use of manipulators in mobile and glove box scenarios (Lopez et al., 2022). The JET fusion reactor uses MASCOT (Skilton et al., 2018), consisting of two 7 degree-of-freedom tele-manipulators for routine inspection and maintenance tasks. MASCOT is mounted on the articulated boom of a telescoping arm (the TARM Burroughes et al. (2018)), which allows it to be moved around the fusion vessel without impacting the sides.

The purpose of DMUs is to extend the capabilities of its users during the operation of a nuclear robot. DMUs can be programmed to reflect the physical status of each component. These Digital Twins are virtual representations of physical objects or processes that are periodically updated to reflect their physical counterpart, for the purpose of mock-up. Digital Twins are primarily used in simulations and data visualisations during the design and development stages, they have since expanded their scope as vital components of cyber-physical systems (Kaigom and Roßmann, 2021), (Douthwaite et al., 2021). In the nuclear industry, the integration of physical robotic systems and their Digital Twins enable an intuitive human-robot interaction, such as the combination of VR and a Leap Motion controller for tele-operating a robotic manipulator Jang et al. (2019) and the use of mixed reality systems for remote inspection (Welburn et al., 2019). DMUs have been used in nuclear fusion engineering in the design of the Wendelstein-7X stellarator fusion device (Renard et al., 2017), and the design verification of ITER's remote handling systems (Sibois et al., 2014).

Digital Twins can also refer to digital environments based on actual locations where a robot may be operated (Blair, 2021) (Jang et al., 2021). The purpose of environmental Digital Twins in robotics is to digitally represent and recreate the physical boundaries and external processes that interact with the robotic device. In this way, a robot and its intended user are provided with accurate information on its surroundings, leading to better and more efficient mission planning (Wright et al., 2021).

It is worth noting that simulations and DMUs, while effective visualisation avenues for robot states, are as effective as how users interpret the data. In the recent years, the implementation of better human-robot interaction strategies continue to grow as this aspect of robot operations and its contribution to the efficiency of a mission is more realised. Such in the case of implementing virtual and augmented reality to robotic systems to increase user immersion through heightened situational awareness and control during the operation (Welburn et al., 2019).

2.3 Deep Learning for Nuclear Industry and DMUs

With the access to advanced hardware and large training dataset, deep learning shows its potential to be used in the nuclear industry to improve production efficiency, reduce operation cost, and improve safety. For long range teleoperation, it is possible to stream only the vision-based detection result instead of transmitting the whole point cloud or real-time videos from the decommissioning site to the operator. VR environments could render the digital representation instead of the raw data that will require a large internet bandwidth, otherwise, data compression and decompression might be needed (Pacheco-Gutierrez et al., 2021). We introduce the application of deep learning in the nuclear industry in three fields, i.e., vision based object detection, sequence data processing, and deep reinforcement learning based control system.

Periodic inspection of the equipment and prediction of Remaining Useful Life (RUL) are common ways for ensuring the safe operation of nuclear industry. As the nuclear environment is complex, with high radiation dose exposure, it is inefficient and expensive to operate manual periodic inspection. A crack detection algorithm based on Naive Bayesian data fusion scheme and CNN for nuclear reactors was proposed in Chen and Jahanshahi (2017). This method enables autonomous detection for each video frame and it achieves 98.3% hit rate against 0.1 false positives per frame. A multi-scale attention mechanism guided knowledge distillation method is proposed in Lang et al. (2021) for surface defect detection. It enables a student model to mimic the complex teacher model through the use of knowledge distillation techniques. A class-weighted cross entropy loss was introduced to address the imbalance of foreground and background in defect detection. The efficient performance of the proposed algorithm was validated by using three benchmarks. Convolution kernel was integrated with Long Short-Term Memory (LSTM) in Wang et al. (2020) for predicting the RUL of electric valves by using the excellent capability of sequential analysis of LSTM.

Sequential data includes text document data and sensor sequence signal data in the nuclear industry chain. The deep learning researchers mainly use Natural Language Processing (NLP) algorithms or LSTM algorithms to process signal data for prediction and classification. Based on NLP techniques, a rule-based expert system, Causal Relationship Identification (CaRI), is proposed in Zhao et al. (2019). The proposed method is applied to analyze the abstract section of the reports from the U.S. Nuclear Regulatory Commission Licensee Event Report database. Based on signal processing technique named cepstral analysis (Jorge et al., 2010), an automatic speech recognition interface is developed to serve as a new operator interface in VR environment for operating virtual control task through spoken commands instead of keyboard and mouse. In Ramgire and Jagdale (2016), a speech control system is developed to control a robotic arm with flexiforce sensor to pick and place objects. Mel-Frequency Cepstrum Coefficients (MFCC) algorithms were introduced to extract features for speech and speaker recognition. The speech recognition can be used for security authentication and speech automatic recognition is used for machine control. The sequential data process using deep learning could improve the operating efficiency while the VR operator is executing missions in VR environment.

Deep reinforcement learning has also become a common method in solving control problems in nuclear applications, because of its efficient computing strength and because it does not require a system model in advance. Deep reinforcement learning and proximal policy optimization are integrated in Radaideh et al. (2021) by establishing a connection through reward shaping between reinforcement learning and the tactics fuel designers follow in practice by moving fuel rods in the assembly to meet specific constraints and objectives. This algorithm is applied on two boiling water reactor assemblies of low-dimensional ($\sim 2 \times 10^6$ combinations) and high-dimensional ($\sim 10^{31}$ combinations) natures. The results demonstrate the proposed algorithm find more feasible patterns, 4-5 times more than Stochastic Optimization (SO), by taking advantage of RL outstanding computational efficiency. Another research work by Park et al. (2022) applied reinforcement learning in Compact Nuclear Simulator (CNS) and key elements for reinforcement learning are designed to be suitable for the heat-up mode. A neural-network structure and a CNS deep RL mechanism are presented as a solution to the automatic control problem. An asynchronous advantage actor-critic algorithm was integrated with a LSTM network to solve the operator task for which establishing clear rules or logic was challenging in (Lee et al., 2020). The proposed neural network was trained using CNS system and was proven capable of identifying an acceptable operating path for increasing the reactor power from 2% to 100% at a specified rate of power increase, and its result was found to be identical to that of the established operation strategy.

This section covered a literature review of the relevant areas of interest for a nuclear decommissioning NG-DMU. The next section will set out the key features identified for an NG-DMU created for a nuclear decommissioning use-case.

3 KEY SOFTWARE FEATURES

The NG-DMU is a culmination of many different technologies and research areas that will all be used to improve the decommissioning process. It can be split into 3 main sections: the simulation, the robotics, and the usability.

A list of desirable features was used to evaluate the simulation software investigated. The assessment is qualitative due to the absence of set standards of measurement for many of the features under consideration.

3.1 Digital Model Simulation features

The Digital Model is a realistic virtual representation of the target environment and is one of the core properties of the Digital Twin. It requires a variety of technologies and disciplines, including: kinematics and dynamics, control, deformation, environmental simulations, radiation simulations, CAD models, control system simulation, and many more.

The following criteria will investigate the feasibility of software by their overall simulation properties: physics engines, rendering functionality, environmental simulations, rigid body dynamics and control and camera/scene properties.

3.1.1 Physics Engines

A physics engine provides an approximation of physical parameters to create a more realistic representation of the scene it is modelling. This work is interested in the range of physics engines available, and the type of engine available - Bullet (E. Coumans and Y. Bai, 2022), ODE (Russ Smith, 2022), PhysX (NVIDIA Corporation, 2022d) for example.

3.1.2 Rendering

The rendering property within a simulation is defined as the process that creates photorealistic 3D model within the scene and includes myriad properties (lighting, shading, texture quality, etc). This work is interested in the rendering engine, and the quality of its output.

3.1.3 Environmental Simulations and Lighting effects

Environmental Simulations (Fluids, heat, radiation etc) are critical for the use case in question as the environmental properties for decommissioning can vary wildly. In this work, this criterion is assessed by the range of environmental simulations that can be simulated relevant to expected use case requirements, and if these simulations can be run natively - within the software and not reliant on a 3rd party plugin for example.

The lighting within 3D software is hugely important to how the user observes the scene that they are operating in and a trade-off is always made between performance optimisation and lighting quality. As a result, the view generation software being used must be capable of editing the lighting effects visible inside the scene extensively. This criteria is interested in the availability of real time lighting, the lighting types and light probes, the customisation and range of precomputation techniques (baking, compositing, caching, etc), colour space (Linear and gamma), simulations beyond standard visual spectrum (IR, etc).

3.1.4 Rigid Body Dynamics and Control

Rigid Body Dynamics are critical for remote handling use cases as they use robotic actuators as an input for the simulation and they must be modelled as accurately as possible. This criteria is interested in:

- Compliance
- Flexibility of joints
- Type and number of surrogate models available
- Real-time vs non-real time characteristic
- Contact interaction (rigid vs impulse)
- Soft body and advanced multi body dynamic packages
- Available API (application programming interface) and plugin options

3.1.5 Camera Properties

Camera Properties relates to how a scene is displayed, navigated and edited. The following criteria will focus on the general in-scene properties that are available within the view generation software:

3.1.5.1 View Control

This is the basic way in which the scene can be viewed within the view generation software. The end user for this NG-DMU will be operations and project engineers and other personnel that require a range of controls to navigate the environment. This criteria is interested in the range of view control options (orbiting, pilot, shortcuts, hardware interfacing, etc), whether the software has a tool to support changing the perspective of the view (isometric and orthogonal viewpoints), whether cross section views are available, and whether objects can be easily centred and the view adjusted.

3.1.5.2 Camera/Scene View Properties

Simulated camera representation is vitally important for remote operation of a manipulator, as it enables the user to adjust their view to provide as much information as possible. For the purpose of this case study, the criterion concerns the native editing of cameras within the scene (FoV, path planning, etc), the ability to view multiple concurrent viewpoints, the effect of multiple cameras on performance, and the integration to real camera hardware (AR, registration, etc).

3.1.5.3 View Customisation

The ability to customize the view of the environment is very interesting, as this can be used to improve the overall quality of the image being viewed. This criteria is interested in the intrinsic camera parameters (real world camera properties, FoV, Aspect Ratio, lens distortion, etc), and the extrinsic camera parameters (pan, tilt, etc)

3.1.5.4 Scene Graph Editing

A scene graph is a generic data structure that is used in view generation software to illustrate the spatial representation of a graphical scene often represented as a tree with the nodes of that tree representing objects. This is the live data structure that stores the objects within the scene and how they relate to each other; editing this graph enables you to change the scene properties.

This criteria is concerned with the availability of object transformation (e.g. translate, rotate, scale), the attachment of objects (i.e. changing object parent), and functionalities such as undo & redo.

3.2 Robotics Features

3.2.1 Virtual Sensors

Having virtual sensors be applied within the simulation will enable the developer to create a more realistic NG-DMU, where users can receive data from the sensors being used in the environment. Examples of sensors that can be used include: LIDAR, IR, Force-Torque, Proximity, Cameras, IMU, etc. For this case study, this criterion concerns: the range of sensors that area available within the simulation, how these sensors can then be further edited, how the sensor information is displayed to the user.

3.2.2 Robot Model Library

It is important that a range of robots can be tested within this simulation to ensure flexibility within the use case environment. As a result, this criterion is interested in: the range and type of robots available within the software (arms, wheeled, locomotion, parallel, etc), how often the libraries are updated, the extent to which the robotic actuators can be edited (different end-effectors for example), and the ease to add CAD models of bespoke robotic hardware with supporting plugins for kinematic representation.

3.2.3 Robotic Specific Features

The remote handling of a robot can be assisted through software certain features within the robot simulation. This criterion is concerned by: the number of features that are available within the software (object detection, learning, training, path planning, locomotion etc), the kinematic movement available (inverse/forward), the data that is displayed in the simulation (HMI) and the DH parameters that can be used. It is also important to consider the ease of which they can be implemented and the extent of the customisation available. It is important to identify what features are available internally/natively and what features can be implemented using an external/plugin. Furthermore, certain features are more relative to the use case in which the NG-DMU would be used.

3.2.4 Haptic Interface

The operator team will be using the software for decommissioning and may require haptic feedback to ensure the remote handling provides them with as much feedback as possible. Therefore, it is important that the robotic simulation software being used can interface with a haptic device. This criterion is measured by the availability of a haptic interface internally within the software, the ease of which this interface can be implemented (this can be an arduous process), the type of feedback that is available, and the customisation of this interface (force ratios, collision parameters, etc)

3.2.5 Deep Learning Capabilities

The capability of supporting deep learning algorithms in simulation is very important as it provides us the opportunity to make robot to learn variety kinds of behaviour in simulation before transferring to the real robot. When robot learns navigation or control policy, it will possibly make mistake or occur operational error, thus, making it learns each behaviour in simulator first will reduce the operational cost and risk significantly. Moreover, it is possible to accelerate the learning process using simulator, as some simulators support simulating multiple robot agents simultaneously, generating a lot of training dataset and allowing each robot to learn parallelly.

3.3 Usability features

The usability of the software is a measure of how easy and intuitive the software is to use. For example, the documentation fidelity, and the import and export processes available. The final criteria bracket focuses on the overall usability, ergonomics and workflow of the system and User experience (UX) of the software, to determine if it is suitable for use in the NG-DMU. To determine this criterion the following sub criteria points have been made: Scene import and export, API and Plugin availability, overall ergonomics and UX, Licensing/Maturity, Documentation and Assistance.

3.3.1 Import and Export/Scene Management

The general workflow for the project must be assessed and compared with supporting APIs and software to determine the validity of the software overall. Considering most decommissioning projects are long-term with multiple collaborating engineers, a shared ergonomic workflow is vital.

An NG-DMU project will likely require multiple engineers, designers and operators to collaborate, potentially internationally, which produces a variety of logistical challenges. Therefore, it is imperative that the operation of this software be as ergonomic as possible and thus the importation and exportation of a scene must be user friendly. This criterion is assessed by: The use of industry standard scene file types for import and export (XML, USD, etc), the quality of the exported scene (lost data, etc), exporting selected objects as part of a scene, version control and collaborative editing of a project. Also of interest is whether using the software result in “vendor lock-in”, where that software must be used exclusively.

3.3.2 Security

Cloud services have now become an established part of modern data storage, and simulation software is no different - however this raises the question of adequate security for sensitive data, and in the case of models uploaded to online libraries, the owner of the model IP. Most of the software presented in this work use files stored locally, and do not require an ongoing internet connection - excepting when accessing online-only content such as model libraries for example. While security of the data should be a consideration for users, an in-depth analysis of encryption and security standards will not be explored in this work.

3.3.3 Documentation & Tools

Although the NG-DMU is designed to be robust and intuitive, the supporting software systems should have a range of technical tools and documentation to assist users. A competent software will have significant support and documentation available with up-to-date wiki entries, a popular forum and tools to diagnose any issues that may occur. This criteria is assessed by the quality of official and/or community documentation to support development, and official tools provided by the distributor that can be used throughout development.

3.3.4 Plugins and API Support

The features and evaluation section of this work is not intended to be exhaustive of all the features required for an NG-DMU. The reviewed software is also unlikely to have all of the functionality that has been mentioned. Therefore, it is vital that the simulation software have a wide range of API functionality and plugin support. This criterion is interested in: The support available for APIs and Plugin modules (documentation, community support, etc), the ease of which these APIs/Plugins can be integrated, whether off-the-shelf API/Plugins for hardware connectivity (haptics, robots, etc) are available, the API/Plugin capability and extensibility available, and the range of APIs/Plugins that are available for any missing criteria in relation to this report.

For a successful NG-DMU, multiple different technologies will need to work together, thus, plugin and API support is vitally important.

3.3.5 Licencing & Maturity

A successfully deployed NG-DMU would be used for several decades and thus any supporting software that will be used must have considerable support. To assess the maturity and longevity of the suggested software(s), the following are considered:

For open-source software: The git commit history (such as number of commits, forks, stars and issues), with emphasis on recent git history (2020-present) and activity (new versions released, forks, commits, etc). This ensures ongoing, community-driven support.

For licensed software: The age of the software being used and reliance on external support.

This section covered the key features of a simulation software used to create an NG-DMU for a nuclear decommissioning use-case.

4 REVIEW & SURVEY

This section will look at the different simulation tools available. It will give an overview of each piece of software, its intended user-base, and the notable features of the software.

4.1 CoppeliaSim (V-Rep)

CoppeliaSim (Coppelia Robotics, Ltd, 2022) is a robotic development toolkit developed by Coppelia Robotics. Previously called V-Rep, the toolkit is opensource with commercial licences available which enables extended functionality, plugin support and integration with other tools.

The software has a distributed control architecture, where each object or model within a scene can be individually controlled - either by an API client, plugin, ROS/ROS2 node (Open Robotics, 2022b), etc. The software can be tailored to bespoke requirements. The controller can be written in several different languages (C, C++, Python, Java, MATLAB (The MathWorks, Inc, 2022), Lua, Octave), resulting in a versatile toolkit.

It supports the Bullet physics library, Open Dynamics Engine (ODE), Vortex Studio (cmlabs, 2022), Newton Dynamics engine (Julio Jerez and Alain Suero, 2022). The user is able to set the physics engine used by the software. This is significant, as the rigid body dynamics customisation is dependent on the physics engine. The rendering quality is high and it supports both simple OpenGL (Khronos Group, 2022) rendering and GPU intensive rendering. While the native library is not extensive, CoppeliaSim supports the AutoCAD (Autodesk Inc, 2022) file format DXF for shape import. The mesh import/export functionality is handled via a plugin. Collision detection is available, along with highlight of collision objects.

Proximity Sensors (customisable ray types, detection volume, etc) area avilable, as well as Vision Sensors (customisable by resolution, API, etc), and Force Sensors (customisable by filters, sample size, trigger settings, etc).

CoppeliaSim supports haptic devices through ROS support, with tutorials on its setup. A Geometric plugin available to enable robotic features implemented independent of the full simulation. Path planning is also implementable.

4.2 Gazebo

Gazebo (Open Source Robotics Foundation, 2022b) is an established and well-known robotic simulation toolkit. It provides a large library of robots and physics engines and a variety of interfaces and virtual sensors for users to design and test robotic solutions. It also has external interfaces capable of working with both ROS and ROS2. It has strong support and version control, with several stable releases being developed over the years.

Four Physics engines are available in Gazebo (ODE, Bullet, Sim-body (Michael Sherman and Peter Eastman, 2022), and DART), which handle rigid body dynamics. Utilizing the OGRE rendering engine, Gazebo provides realistic rendering of environments including high-quality lighting, shadows, and textures. Fluid simulation is also available.

Gazebo has extensive sensor, robot and actuator libraries from laser range finders (Niu et al., 2021a), 2D/3D cameras, Kinect-style sensors (Microsoft, 2022), contact sensors, force-torque. Many robots are provided including PR2 (Manny Ojigbo, 2014), Pioneer2 DX (Cyberbotics Ltd., 2022), iRobot Create (iRobot Corp, 2022), Universal robot arm (Liu et al., 2021), Kuka robot arm (Niu et al., 2021b) and TurtleBot (Open Source Robotics Foundation, Inc, 2022) (Lin et al., 2021). Comparing with mobile robot and robotic arm, unmanned marine vehicle is more challenging to be simulated as it takes into account the dynamics of wind, wave, and sea current as well to help design the energy efficient control algorithm (Niu et al., 2016) (Niu et al., 2018) (Niu et al., 2020) (Niu et al., 2017) instead of just path length optimized algorithm (Niu et al., 2019) (Lu et al., 2016). Thanks to the powerful dynamics simulation engine of Gazebo, it also supports unmanned marine vehicle simulation Manhães et al. (2016) that has the ROS API as well. Moreover, Gazebo provides the functionality of supporting multiple mobile robots Hu et al. (2020) Na et al. (2022) and multiple robotic arms. Gazebo also facilitates object detection and HAPTIX (Hand Proprioception & Touch Interfaces) (Defense Advanced Research Projects Agency, 2015).

The RAIN research hub has used Gazebo to assess ionising radiation levels in nuclear inspection challenges (Wright et al., 2021).

4.3 Ignition

Ignition (Open Robotics, 2022c) was created as a spin off from Gazebo classic. It is a set of open source libraries Open Source Robotics Foundation (2022a) that encompass the essentials needed for robotic simulation. It facilitates the integration into other services such as ROS/ROS2 for features that are not included natively: e.g. sensor integration, custom plugins, etc. Its goal is to combine the usability and variety available in Gazebo with a modular, plugin based approach - moving away from Gazebo's monolithic architecture.

Ignition uses the DART - Dynamic Animation and Robotics Toolkit (Lee et al., 2018) - physics engine by default, however it does allow the user to choose a different engine, if desired. It has a similar approach for rendering engines, and supports OGRE (Ogre3D Team, 2022) and OptiX (NVIDIA Corporation, 2022c).

Models in Ignition can be loaded from SDF file format. Ignition supports collision shapes, such as box, sphere, cylinder, mesh, and heightmap. Joint types supported include fixed, ball, screw, and revolute. It can carry out step simulations, get and set states, as well as apply inputs.

Ignition Senors is an open source library that provides a set of sensor and noise models accessible through a C++ interface. Sensors include monocular cameras, depth cameras, LIDAR, IMU, contact, altimeter, and magnetometer sensors. Each sensor can optionally utilize a noise model to inject Gaussian or custom noise properties. The library aims to generate realistic sensor data suitable for use in robotic applications and simulation.

4.4 Nvidia Omniverse Isaac Sim

Nvidia Omniverse Isaac Sim (NVIDIA Corporation, 2022b) is robotic simulation tool launched in 2020 which aims to simplify the entire pipeline for developing robotic simulations. It aims to capitalise on the RTX GPU's (NVIDIA Corporation, 2022a) computing capability for simulations and rendering. Nvidia Omniverse Isaac Sim uses the latest version of PhysX, and has the full suite of Nvidia rendering tools, and access to other rendering tools as well. Omniverse Flow is available for fluid simulations, smoke simulations, and customisable particle emitters for configurable simulations. Isaac Sim does have rigid body dynamics, and it also supports the Omniverse connect system for external plugins. It integrates with other industry standard tools (ROS/ROS2, Maya (Autodesk Inc., 2022), SOLIDWORKS (Dassault Systèmes SolidWorks Corporation, 2022), Unreal 4 (Epic Games, Inc, 2022), etc) through the Omniverse Nucleus. While the inter-connectivity of services such as is very attractive for collaborative purposes, it does introduce the issue of 'vendor lock in', where the user is committed to a specific software solution, as switching from the product is impractical.

4.5 WeBots

WeBots (Cyberbotics Ltd, 2022b) is an opensource, multi-platform desktop application used to simulate and build robotic solutions. Developed by Cyberbotics Ltd, the software is straightforward, easy to use, and has use cases in the education and research sectors (Cyberbotics Ltd, 2022a). The features available in Webots are simple, powerful and provide good customisation options using the QT GUI for editing and OpenGL 3.3 for rendering. Development can be done using C, C++, Python, Java, MATLAB or through ROS with API integration.

Webots uses the ODE (Open Dynamics Engine) for collision detection and rigid body dynamics simulation. The ODE library provides accurate simulation of objects' physical properties, such as velocity, inertia and friction.

The following sensors are supported by Webots: Distance Sensor, Range Finder, Light Sensor, Touch Sensor, Inclinometer, Compass, and Camera. Users can model a linear camera, a typical RGB camera or even a fish eye which is spherically distorted. The virtual camera images can be displayed on a VR headset device such as the Oculus Rift (Facebook Technologies, LLC., 2022), or HTC Vive (HTC Corporation, 2022).

Webots also provides access to the large Webots asset library which includes drones (Alsayed et al., 2021) (Alsayed et al., 2022), mobile robots (Ban et al., 2021), sensors, actuators, objects, and materials.

4.6 Choreonoid

Choreonoid (Nakaoka, 2012) is an extensible virtual robot environment developed by the National Institute of Advanced Industrial Science and Technology (AIST) in Japan. Its main attraction is its extensibility with other frameworks and software solutions. Choreonoid applies OpenGL3.3 for rendering engine, and it supports 4 different physics engines: the Bullet physics library, Open Dynamics Engine (ODE), PhysX Engine (NVIDIA Corporation, 2022d), and AGX Dynamics (Algoryx Simulation AB, 2022b).

Choreonoid AGX Dynamics plugin provides the ability of real time simulation of a crawler robot, wires or other functions. Users can change camera parameters, and the following sensors are supported: Range Finder, Range Camera, Light Sensor, Force Sensor, Gyro Sensor, Acceleration Sensor. Choreonoid has a graspPlugin that can be used to solve problems such as grasp planning, trajectory planning and task planning.

Remote decommissioning tasks using a remotely operated robot can be simulated using the HAIROWorldPlugin, which provides simulation functions such as Fluid dynamics, Camera image generator effects (such as distortion, Gaussian noise, colour filter, and transparency), Communication failure emulator, etc.

4.7 AGX Dynamics by Algoryx

AGX Dynamics (Algoryx Simulation AB, 2022b) is a Software Development Kit (SDK) for modelling and simulation of mechanical systems. It is a multi-purpose physics engine, and includes contacts and friction that can be used as either a Unity or Unreal integrated package or extended to a bespoke piece of software via their Software Development Kit. The software comprises of a core library of basic functionality that includes rigid bodies, joints, motors, automatic contact detection and much more; delivering high fidelity, stability, and speed. The 2 “off the shelf” interfaces available for AGX Dynamics are Unity and

Unreal Engine 4, however the potential for developing a bespoke solution is possible using the SDK and customer support. More detailed sub modules can be attached to the SDK depending on the user requirements, however only a portion of these sub modules are included in the Unreal integration of AGX Dynamics.

4.8 Modular Open Robots Simulation Engine (MORSE)

Modular Open Robots Simulation Engine (MORSE) (Echeverria et al., 2011) is an academic python-based simulator for robotics. It can simulate realistic 3D environments using the Blender game engine.

As it is an academic project, it is developed on Linux and there is limited support for MacOSX or Windows. Support is limited to documentation and user-forums.

4.9 VR4Robots

VR4Robots (version 12) is a proprietary commercial VR system from Tree C technology (Tree-C, 2022). The older version 7 has been used for remote handling at JET for several years. There are two configurations for VR4Robots: a kinematic system, and a dynamic system. The kinematic system uses control system data to animate virtual machines. The dynamic system provides a physics engine to simulate physics processes, and must be tailored to the specific environment. The software is mature and established in the remote-handling market. The systems and functionality are designed around using robots. It has suitable inverse kinematics. Scenes can be connected to a network and controlled or viewed from multiple PCs.

The UI/UX design is reliant on dual monitors with no 4K support. Customisation is difficult to implement by the user. However, these can instead be requested as additional features to the core software when negotiating the software licence. The virtual environment must be imported from 3dsMax, requiring a separate license. Reliance on 3DSMax can create issues with versioning. New 3DSMax versions are released annually, but VR4R is not similarly updated, leading to reliance on outdated software. The proprietary model format (.vmx) has no export capabilities, leading to vendor lock-in. There is limited documentation, however paid training courses are available. The user is reliant on a support contract to fix software bugs.

4.10 RoboDK

RoboDK is an offline programming and simulation software with an extensive library of kinematic robot models. Its standard interface requires no programming experience, and it is easily extensible through APIs in Python, C# and Matlab. It also has detailed documentation for both the basic functionality as well as API support. Plugins are available for popular CAD/CAM software such as SolidWorks, Fusion 360, and Inventor.

It provides the ability to communicate with physical robot systems, and upload robot programs generated from an offline simulation.

However, it has no Physics engine, and low quality textures and lighting. The collision mapping is not accurate, and the CAD/CAM functionality is basic and slow with larger models. RoboDK does offer Inverse Kinematics, however the documentation specifies that the simulated movement may not be the same as the actual movement, and does not offer rigid body dynamics.

The robot library is extensive, however custom robots are difficult to add.

4.11 Toia

Toia is a software library developed for haptic rendering and supporting multiple devices (6 DoF robotic arms with haptic features, multi-finger haptic gloves, ultrasonic haptic arrays, etc.) to provide appropriate haptic feedback from a DMU. The platform integrated with Carbon physics engine supporting simulation of soft and rigid materials with collusion and motion constraints (e.g., joints of robotic manipulator) for the real-time haptic simulation. Toia utilizes the Unreal Engine 4 (UE4), which brings a full suite of 3D authoring and visualization tools, as a primary front end for the development of haptic simulation. To enhance performance, the platform separate haptic and graphical fidelity for deformable objects: haptic physics meshes are lower in polygon count than their respective visual counterparts.

This section covered the different simulation tools being reviewed in this paper. It presented an overview of each software, the main user-bases, and some notable features of interest.

5 DISCUSSION

Digital mock-ups and digital twins are fast growing areas in the development and implementation of nuclear decommissioning activities. The availability and presentation of various simulation tools and libraries will provide a go-to guide in industrial applications for industry professionals and will also contribute to the increase in the number of new publications to be produced by academics in this field.

This paper provides an overview of different simulation software tools that have the potential in the expanding robotics field for nuclear environments. Firstly, virtual reality, digital twin, and deep learning solutions for the nuclear industry and DMUs are discussed by investigating the state-of-the-art. Then, we identified a necessary list of assessment criteria features for evaluating each of the simulation tools in terms of simulation, robotics, and usability details by analysing the state-of-the-art challenges of solutions. After that, we examined the simulation tools by analysing their particular characteristics in three-stage concepts. The Tables 1, 2, and 3 show how the existing simulation software compares and the different capabilities they offer in each of the areas of interest.

Each of the software tools presented in this paper has its own set of features. Regarding simulation features, all but one (RoboDK) of them provide physics engines and rendering capabilities. In particular, CoppeliaSim and Nvidia Omniverse Isaac Sim support GPU intensive rendering, which can be useful for robotics applications that require heavy computations. Most of the softwares investigated offer environmental simulation beyond lighting effects and camera options. While some of them additionally include water, fog, light, and light simulations, at least half of them have fluid simulations. The rigid body dynamics feature that is crucial in remote handling scenarios is present in the majority of the software.

With respect to robotics features, a large number of software tools offer at least some functionality for the integration of virtual sensors in which the sensor support is a valuable factor in supporting real-world conditions. However, the types of sensors they support are variable and comparatively few support a haptics interface. A number of commonly used sensors, such as vision and force sensors, especially for remote handling applications, are included in simulation tools such as CoppeliaSim, Gazebo, and Choreonoid.

Another important factor to consider is the ability to import digital models and scenes, which allows for the simulation of nuclear environments by transferring experience from various off-the-shelf drawing software. With the exception of VR4Robots, the others have either internal robot models or integrations to import robot and shape models from many popular third-party file formats such as CAD, DXF, STL, COLLADA, URDF, and many more.

Deep learning capability is also a beneficial consideration and can provide AI-based learning and predictive and preventive decision making in a wide range of decommissioning tasks. All reviewed simulation tools are qualified to develop deep learning algorithms using Python API integration, with the exception of VR4Robots. Furthermore, the virtual reality integration feature can help with an immersive user experience by simulating task demonstration and inspection. For viewing the simulation and human-robot interaction modalities, the virtual reality headsets HTC Vive and Oculus Rift are supported by Webots.

Finally, usability and documentation are very variable. Although commercially licensed software products have comprehensive online documentation, the others have either an online community or GitHub documentation, or both. Furthermore, some only support one programming language API plugin, while others support multiple programming language API plugins, such as C, C++, C#, Python, Matlab, Java, and Lua.

Table 1. Comparison table of the simulation features available in the different software reviewed in this work

Criteria ID	Physics Engine(s)	Rendering	Environment Simulations	Rigid Body Dynamics	Camera Properties
CoppeliaSim CoppeliaSim (2022)	✓	✓	✓	Dependent on the physics engine.	✓
Gazebo Open Source Robotics Foundation (2022c)	✓	✓	Fluid simulations	✓	✓
Ignition Open Robotics (2022a)	✓	✓	✗	✓	✓
Nvidia Omniverse Isaac Sim NVIDIA (2022b)	✓	✓	✓	✓	✓
Webots Ltd. (2022)	✓	✓	✓	✓	✓
Choreonoid Choreonoid (2022)	✓	✓	✓	✓	✓
AGX Dynamics Algoryx Simulation AB (2022a)	✓	✓	✓	✓	✓
MORSE MORSE (2022)	✓	✓	Environment modelling can be created in Blender then imported	✓	✓
VR4Robots Tree-C (2022)	Available as part of the dynamics package	✓	✓	Included in the dynamics package	✓
RoboDK Inc. (2022)	✗	OpenGL customizable shaders are available	✗	✗	✓
Toia Generic Robotics (2022)	✓	✓	✓	✓	✓

Table 2. Comparison table of the robotics features available in the different software reviewed in this work

Criteria ID	Virtual Sensors	Robotic Model Library	Robotic Features	Haptic Interface	Deep Learning capabilities
CoppeliaSim CoppeliaSim (2022)	✓	Limited native library. Large number of file formats supported for import.	✓	Supported through ROS	✓
Gazebo Open Source Robotics Foundation (2022c)	✓	✓	✓	Available using HAPTIX packages	✓
Ignition Open Robotics (2022a)	✓	✓	✓	✗	APIs can deploy deep learning algorithms
Nvidia Omniverse Isaac Sim NVIDIA (2022b)	✓	✓	✓	✗	✓
Webots Ltd. (2022)	✓	✓	✓	✗	✓
Choreonoid Choreonoid (2022)	✓	✓	✓	✗	APIs can deploy deep learning algorithms
AGX Dynamics Algoryx Simulation AB (2022a)	Supported through the unity plugin Kallin (2019)	✗	✓	✗	✓
MORSE MORSE (2022)	✓	✓	✓	✗	✓
VR4Robots Tree-C (2022)	✓	✗	✗	✗	✗
RoboDK Inc. (2022)	✓	✓	✓	✗	✓
Toia Generic Robotics (2022)	Supported using ROS/ROS2 plugin	✓	✓	✓	✓

Table 3. Comparison table of the usability of the different software reviewed in this work

Criteria ID	Import / Export	API/Plugins	Licensing	Documentation
CoppeliaSim CoppeliaSim (2022)	File import: OBJ, DXF, STL, DAE, URDF File export: OBJ, STL, DAE. Support for heightfield data :formats include JPEG, PNG, TGA, BMP, TIFF, GIF file, CSV, TXT	✓	GNU GPL Source code + Binary licensing (commercial license or free educational license). Other plugins have a BSD license	User manual and online forum available. Github available.
Gazebo Open Source Robotics Foundation (2022c)	File import: DAE, STL, OBJ, SVG	✓	Apache License, Version 2.0	Online community, online tutorials and Github available.
Ignition Open Robotics (2022a)	File import: STL, OBJ, DAE, SVG, BVH. File export: DAE	✓	Apache License, Version 2.0	Online User manual
Nvidia Omniverse Isaac Sim NVIDIA (2022b)	File import: FBX, OBJ and GLTF. Using extenstions: STEP, IGES, and URDF	✓	Nvidia Omniverse License Agreement NVIDIA (2022a)	Online documentation available.
Webots Ltd. (2022)	File import: DAE, STL, OBJ. File export: WRL	✓	Apache License, Version 2.0	Online User Guide
Choreonoid Choreonoid (2022)	File import: WRL	✓	MIT	Online User manual and user forum
AGX Dynamics Algoryx Simulation AB (2022a)	Extensive list of file formats supported, incl STL, OBJ, DAE, FBX, URDF. Export of Functional Mock-up Interface available FMI Standard (2022)	Matlab/Simulink plugin (Windows only)	License required for either development or runtime for deployment	Online documentation and tutorials
MORSE MORSE (2022)	File import: Blender files (low-poly) File export: Requires add-on.	✓	Permissive BSD license	Github community
VR4Robots Tree-C (2022)	File import: 3DS Max VMX files	✓	Bespoke license, annual subscription	User manual and paid training available.
RoboDK Inc. (2022)	File import (static only): STEP, IGES	✓	Commercial License, annual subscription	Online documentation.
Toia Generic Robotics (2022)	File import: formats supported by Unreal Engine. File export: not supported	✓	Requires a license for both development and deployment	Customer support

6 CONCLUSION

This paper introduced and assessed eleven different simulation software solutions, using criteria identified as important for the creation of a remote-handling Next Generation Digital Mock-up application in the nuclear sector. Simulation and rendering capabilities were well served across each of the concerned software, however the inclusion of haptics and robotics features are more limited. Each software reviewed in this work offers use-case specific solutions, with the functionality offered tailored to their expected application. As expected, there is no single solution that offers the range of requirements for a remote-handling DMU out-of-the-box, however both Gazebo and Toia offer haptics and include the most features highlighted in this paper. They both offer API/Plugin features, however Toia documentation is limited.

ACKNOWLEDGMENTS

This research was fully funded within the LongOps programme by UKRI under the Project Reference 107463, NDA, and TEPCO. The views and opinions expressed herein do not necessarily reflect those of the organizations.

REFERENCES

iRobot Corp (2022). Create 2 robot. Accessed: 2022-02-08

Adamov, Y. O. and Yegorov, Y. A. (1987). Development of robotic systems for nuclear applications including emergencies. *American Nucl. Soc., Int. Mtg on Remote Systems and Robotics in Hostile Environments, Pasco, WA*

Algoryx Simulation AB (2022a). About agx dynamics for unity. Accessed: 2022-04-01

Algoryx Simulation AB (2022b). Agx dynamics real-time multi-body simulation. Accessed: 2022-02-08

Alsayed, A., Nabawy, M. R., Yunusa-Kaltungo, A., Quinn, M. K., and Arvin, F. (2021). An autonomous mapping approach for confined spaces using flying robots. In *Annual Conference Towards Autonomous Robotic Systems* (Springer), 326–336

Alsayed, A., Nabawy, M. R., Yunusa-Kaltungo, A., Quinn, M. K., and Arvin, F. (2022). Real-time scan matching for indoor mapping with a drone. In *AIAA SCITECH 2022 Forum*. 0268

Autodesk Inc (2022). Autocad: Cad software with design automation plus toolsets, web and mobile apps. Accessed: 2022-02-08

Autodesk Inc. (2022). Maya 3d computer animation, modelling, simulation and rendering software. Accessed: 2022-02-08

Ban, Z., Hu, J., Lennox, B., and Arvin, F. (2021). Self-organised collision-free flocking mechanism in heterogeneous robot swarms. *Mobile Networks and Applications* , 1–11

Barbosa, F. S., Lacerda, B., Duckworth, P., Tumova, J., and Hawes, N. (2021). Risk-aware motion planning in partially known environments. In *60th IEEE Conference on Decision and Control, CDC 2021, Austin, TX, USA, December 14-17, 2021* (IEEE), 5220–5226. doi:10.1109/CDC45484.2021.9683744

Blair, G. S. (2021). Digital twins of the natural environment. *Patterns* 2, 100359. doi:https://doi.org/10.1016/j.patter.2021.100359

Blue Robotics (2022). Bluerov2 - affordable and capable underwater rov. Accessed: 2022-03-22

Burroughes, G., Keep, J., Goodliffe, M., Middleton-Gear, D., Kantor, A., Clark, E., et al. (2018). Precision control of a slender high payload 22 dof nuclear robot system: Tarm re-ascending - 18346

Cheah, W., Khalili, H. H., Arvin, F., Green, P., Watson, S., and Lennox, B. (2019). Advanced motions for hexapods. *International Journal of Advanced Robotic Systems* 16, 1729881419841537. doi:10.1177/1729881419841537

Chen, F.-C. and Jahanshahi, M. R. (2017). Nb-cnn: Deep learning-based crack detection using convolutional neural network and naïve bayes data fusion. *IEEE Transactions on Industrial Electronics* 65, 4392–4400

Choreonoid (2022). Choreonoid operation manual. Accessed: 2022-04-01

cmlabs (2022). Vortex studio. Accessed: 2022-02-08

Connor, D. T., Wood, K., Martin, P. G., Goren, S., Megson-Smith, D., Verbelen, Y., et al. (2020). Corrigendum: Radiological mapping of post-disaster nuclear environments using fixed-wing unmanned aerial systems: A study from chornobyl. *Frontiers Robotics AI* 7, 30. doi:10.3389/frobt.2020.00030

Coppelia Robotics, Ltd (2022). Coppeliasim: from the creators of v-rep. Accessed: 2022-02-08

CoppeliaSim (2022). Coppeliasim user manual. Accessed: 2022-04-01

Cryer, A., Kapellmann-Zafra, G., Abrego-Hernandez, S., Marin-Reyes, H., and French, R. (2019). Advantages of virtual reality in the teaching and training of radiation protection during interventions in harsh environments. In *2019 24th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA)* (IEEE). doi:10.1109/etfa.2019.8869433

Cyberbotics Ltd. (2022). Adept's pioneer 2. Accessed: 2022-02-08

Cyberbotics Ltd (2022a). User guide introduction to webots. Accessed: 2022-04-25

Cyberbotics Ltd (2022b). Webots open source robot simulator. Accessed: 2022-02-08

Dassault Systèmes SolidWorks Corporation (2022). Solidworks. Accessed: 2022-02-08

Defense Advanced Research Projects Agency (2015). Haptix starts work to provide prosthetic hands with sense of touch. Accessed: 2022-02-08

Douthwaite, J., Lesage, B., Gleirscher, M., Calinescu, R., Aitken, J. M., Alexander, R., et al. (2021). A modular digital twinning framework for safety assurance of collaborative robotics. *Frontiers in Robotics and AI* 8. doi:10.3389/frobt.2021.758099

E. Coumans and Y. Bai (2022). Bullet real-time physics simulation. Accessed: 2022-02-08

Echeverria, G., Lassabe, N., Degroote, A., and Lemaignan, S. (2011). Modular open robots simulation engine: Morse. In *2011 IEEE International Conference on Robotics and Automation*. 46–51. doi:10.1109/ICRA.2011.5980252

Epic Games, Inc (2022). Unreal engine. Accessed: 2022-02-08

Facebook Technologies, LLC. (2022). Meta quest. Accessed: 2022-02-08

FMI Standard (2022). Functional mock-up interface. Accessed: 2022-04-04

Gazzotti, S., Ferlay, F., Meunier, L., Viudes, P., Huc, K., Derkazarian, A., et al. (2021). Virtual and augmented reality use cases for fusion design engineering. *Fusion Engineering and Design* 172, 112780. doi:https://doi.org/10.1016/j.fusengdes.2021.112780

Gelhaus, F. E. and Roman, H. T. (1990). Robot applications in nuclear power plants. *Progress in Nuclear Energy* 23, 1–33. doi:https://doi.org/10.1016/0149-1970(90)90012-T

Generic Robotics (2022). Generic robotics - feeling is believing. Accessed: 2022-04-28

Groves, K., Hernandez, E., West, A., Wright, T., and Lennox, B. (2021). Robotic exploration of an unknown nuclear environment using radiation informed autonomous navigation. *Robotics* 10. doi:10.3390/robotics10020078

Groves, K., West, A., Gornicki, K., Watson, S., Carrasco, J., and Lennox, B. (2019). Mallard: An autonomous aquatic surface vehicle for inspection and monitoring of wet nuclear storage facilities. *Robotics* 8. doi:10.3390/robotics8020047

HTC Corporation (2022). Vive. Accessed: 2022-02-08

Hu, J., Niu, H., Carrasco, J., Lennox, B., and Arvin, F. (2020). Voronoi-based multi-robot autonomous exploration in unknown environments via deep reinforcement learning. *IEEE Transactions on Vehicular Technology* 69, 14413–14423

Hyun, D., Kim, I., Lee, J., Kim, G.-H., Jeong, K.-S., Choi, B. S., et al. (2017). A methodology to simulate the cutting process for a nuclear dismantling simulation based on a digital manufacturing platform. *Annals of Nuclear Energy* 103, 369–383. doi:https://doi.org/10.1016/j.anucene.2017.01.035

Inc., R. (2022). Robodk basic guide. Accessed: 2022-04-01

Jang, I., Carrasco, J., Weightman, A., and Lennox, B. (2019). Intuitive bare-hand teleoperation of a robotic manipulator using virtual reality and leap motion. In *Towards Autonomous Robotic Systems*, eds. K. Althoefer, J. Konstantinova, and K. Zhang (Cham: Springer International Publishing), 283–294

Jang, I., Niu, H., Collins, E. C., Weightman, A., Carrasco, J., and Lennox, B. (2021). Virtual kinesthetic teaching for bimanual telemanipulation. In *2021 IEEE/SICE International Symposium on System Integration (SII)* (IEEE), 120–125

Jin-Yang, L., Long, G., Zhi-Yong, H., You-Peng, Z., Hu-Shan, X., Yang, S., et al. (2020). The development and application of digital refueling mock-up for china initiative accelerator driven system. *Progress in Nuclear Energy* 127, 103433. doi:https://doi.org/10.1016/j.pnucene.2020.103433

Jorge, C. A. F., Mól, A. C. A., Pereira, C. M. N., Aghina, M. A. C., and Nomiya, D. V. (2010). Human-system interface based on speech recognition: application to a virtual nuclear power plant control desk. *Progress in Nuclear Energy* 52, 379–386

Julio Jerez and Alain Suero (2022). Newton dynamics. Accessed: 2022-02-08

Kaigom, E. G. and Roßmann, J. (2021). Value-driven robotic digital twins in cyber–physical applications. *IEEE Transactions on Industrial Informatics* 17, 3609–3619. doi:10.1109/TII.2020.3011062

Kallin, N. (2019). Sensor simulation is-agxunity a viable platform for adding synthetic sensors

Khronos Group (2022). Opengl. the industry's foundation for high performance graphics. Accessed: 2022-02-08

Lang, J., Tang, C., Gao, Y., and Lv, J. (2021). Knowledge distillation method for surface defect detection. In *International Conference on Neural Information Processing* (Springer), 644–655

Lee, D., Arigi, A. M., and Kim, J. (2020). Algorithm for autonomous power-increase operation using deep reinforcement learning and a rule-based system. *IEEE Access* 8, 196727–196746

Lee, J., Grey, M. X., Ha, S., Kunz, T., Jain, S., Ye, Y., et al. (2018). Dart: Dynamic animation and robotics toolkit. *Journal of Open Source Software* 3, 500. doi:10.21105/joss.00500

Lin, F., Ji, Z., Wei, C., and Niu, H. (2021). Reinforcement learning-based mapless navigation with fail-safe localisation. In *Annual Conference Towards Autonomous Robotic Systems* (Springer), 100–111

Liu, W., Niu, H., Mahyuddin, M. N., Herrmann, G., and Carrasco, J. (2021). A model-free deep reinforcement learning approach for robotic manipulators path planning. In *2021 21st International Conference on Control, Automation and Systems (ICCAS)* (IEEE), 512–517

Liu, Y.-k., Chen, Z.-t., Chao, N., Peng, M.-j., and Jia, Y.-h. (2022). A dose assessment method for nuclear facility decommissioning based on the combination of cad and point-kernel method. *Radiation Physics and Chemistry* 193, 109942. doi:https://doi.org/10.1016/j.radphyschem.2021.109942

Lopez, E., Nath, R., and Herrmann, G. (2022). Semi-autonomous grasping for assisted glovebox operations. doi:10.5281/zenodo.6353535. Second fund from EP/W001128/1, Robotics and Artificial Intelligence for Nuclear Plus (RAIN+)

Ltd., C. (2022). Webots user guide. Accessed: 2022-04-01

Lu, Y., Niu, H., Savvaris, A., and Tsourdos, A. (2016). Verifying collision avoidance behaviours for unmanned surface vehicles using probabilistic model checking. *IFAC-PapersOnLine* 49, 127–132

Manhães, M. M. M., Scherer, S. A., Voss, M., Douat, L. R., and Rauschenbach, T. (2016). Uuv simulator: A gazebo-based package for underwater intervention and multi-robot simulation. In *OCEANS 2016 MTS/IEEE Monterey* (IEEE), 1–8

Manny Ojigbo (2014). Clearpath welcomes pr2 to the family. Accessed: 2022-02-09

Michael Sherman and Peter Eastman (2022). Simbody: Multibody physics api. Accessed: 2022-02-08

Microsoft (2022). Kinect for windows. Accessed: 2022-02-08

MORSE (2022). The morse simulator documentation. Accessed: 2022-04-01

Na, S., Niu, H., Lennox, B., and Arvin, F. (2022). Bio-inspired collision avoidance in swarm systems via deep reinforcement learning. *IEEE Transactions on Vehicular Technology*

Nakaoka, S. (2012). Choreonoid: Extensible virtual robot environment built on an integrated gui framework. In *2012 IEEE/SICE International Symposium on System Integration (SII)*. 79–85. doi:10.1109/SII.2012.6427350

Nancekievill, M., Jones, A. R., Joyce, M. J., Lennox, B., Watson, S., Katakura, J., et al. (2018). Development of a radiological characterization submersible rov for use at fukushima daiichi. *IEEE Transactions on Nuclear Science* 65, 2565–2572. doi:10.1109/TNS.2018.2858563

Nash, B., Walker, A., and Chambers, T. (2018). A simulator based on virtual reality to dismantle a research reactor assembly using master-slave manipulators. *Annals of Nuclear Energy* 120, 1–7. doi:https://doi.org/10.1016/j.anucene.2018.05.018

Niu, H., Ji, Z., Arvin, F., Lennox, B., Yin, H., and Carrasco, J. (2021a). Accelerated sim-to-real deep reinforcement learning: Learning collision avoidance from human player. In *2021 IEEE/SICE International Symposium on System Integration (SII)* (IEEE), 144–149

Niu, H., Ji, Z., Savvaris, A., and Tsourdos, A. (2020). Energy efficient path planning for unmanned surface vehicle in spatially-temporally variant environment. *Ocean Engineering* 196, 106766

Niu, H., Ji, Z., Zhu, Z., Yin, H., and Carrasco, J. (2021b). 3d vision-guided pick-and-place using kuka lbr iiwa robot. In *2021 IEEE/SICE International Symposium on System Integration (SII)* (IEEE), 592–593

Niu, H., Lu, Y., Savvaris, A., and Tsourdos, A. (2016). Efficient path following algorithm for unmanned surface vehicle. In *OCEANS 2016-Shanghai* (IEEE), 1–7

Niu, H., Lu, Y., Savvaris, A., and Tsourdos, A. (2018). An energy-efficient path planning algorithm for unmanned surface vehicles. *Ocean Engineering* 161, 308–321

Niu, H., Savvaris, A., and Tsourdos, A. (2017). Usv geometric collision avoidance algorithm for multiple marine vehicles. In *OCEANS 2017-Anchorage* (IEEE), 1–10

Niu, H., Savvaris, A., Tsourdos, A., and Ji, Z. (2019). Voronoi-visibility roadmap-based path planning algorithm for unmanned surface vehicles. *The Journal of Navigation* 72, 850–874

NVIDIA (2022a). Nvidia omniverse license agreement. Accessed: 2022-10-11

NVIDIA (2022b). What is isaac sim? Accessed: 2022-04-01

NVIDIA Corporation (2022a). Geforce rtx 20 series. Accessed: 2022-02-08

NVIDIA Corporation (2022b). Nvidia isaac sim. Accessed: 2022-02-08

NVIDIA Corporation (2022c). Nvidia optix™ ray tracing engine. Accessed: 2022-02-08

NVIDIA Corporation (2022d). Nvidia physx system software. Accessed: 2022-02-08

Ogre3D Team (2022). Ogre. Accessed: 2022-02-08

Open Robotics (2022a). Ignition documentation. Accessed: 2022-04-01

Open Robotics (2022b). Ros - robot operating system. Accessed: 2022-02-08

Open Robotics (2022c). Simulate before you build. Accessed: 2022-02-08

Open Source Robotics Foundation (2022a). About gazebo. Accessed: 2022-04-25

Open Source Robotics Foundation (2022b). Gazebo: Robot simulation made easy. Accessed: 2022-02-08

Open Source Robotics Foundation (2022c). Gazebo tutorials. Accessed: 2022-04-01

Open Source Robotics Foundation, Inc (2022). Turtlebot3. Accessed: 2022-02-08

Pacheco-Gutierrez, S., Niu, H., Caliskanelli, I., and Skilton, R. (2021). A multiple level-of-detail 3d data transmission approach for low-latency remote visualisation in teleoperation tasks. *Robotics* 10, 89

Park, J., Kim, T., Seong, S., and Koo, S. (2022). Control automation in the heat-up mode of a nuclear power plant using reinforcement learning. *Progress in Nuclear Energy* 145, 104107

Radaideh, M. I., Wolverton, I., Joseph, J., Tusar, J. J., Otgonbaatar, U., Roy, N., et al. (2021). Physics-informed reinforcement learning optimization of nuclear assembly design. *Nuclear Engineering and Design* 372, 110966

Ramgire, J. B. and Jagdale, S. (2016). Speech control pick and place robotic arm with flexiforce sensor. In *2016 International Conference on Inventive Computation Technologies (ICICT)* (IEEE), vol. 2, 1–5

Renard, S., Holtz, A., Baylard, C., and Banduch, M. (2017). Software tool solutions for the design of w7-x. *Fusion Engineering and Design* 123, 133–136. doi:https://doi.org/10.1016/j.fusengdes.2017.04.027. Proceedings of the 29th Symposium on Fusion Technology (SOFT-29) Prague, Czech Republic, September 5-9, 2016

Russ Smith (2022). Open dynamics engine. Accessed: 2022-02-08

Sibois, R., Määttä, T., Siuko, M., and Mattila, J. (2014). Using digital mock-ups within simulation lifecycle environment for the verification of iter remote handling systems design. *IEEE Transactions on Plasma Science* 42, 698–702. doi:10.1109/TPS.2014.2298877

Skilton, R., Hamilton, N., Howell, R., Lamb, C., and Rodriguez, J. (2018). Mascot 6: Achieving high dexterity tele-manipulation with a modern architectural design for fusion remote maintenance. *Fusion Engineering and Design* 136, 575–578. doi:https://doi.org/10.1016/j.fusengdes.2018.03.026. Special Issue: Proceedings of the 13th International Symposium on Fusion Nuclear Technology (ISFNT-13)

The MathWorks, Inc (2022). Math. graphics. programming. Accessed: 2022-02-08

Tree-C (2022). Vr4robots® – the real time interactive visualization technology for remote handling. Accessed: 2022-10-12

Verbelen, Y., Martin, P. G., Ahmad, K., Kaluvan, S., and Scott, T. B. (2021). Miniaturised low-cost gamma scanning platform for contamination identification, localisation and characterisation: A new instrument in the decommissioning toolkit. *Sensors* 21, 2884. doi:10.3390/s21082884

Verbelen, Y., Megson-Smith, D., Russell-Pavier, F. S., Martin, P. G., Connor, D. T., Tucker, M. R., et al. (2022). A flexible power delivery system for remote nuclear inspection instruments. In *8th International Conference on Automation, Robotics and Applications, ICARA 2022, Prague, Czech Republic, February 18-20, 2022* (IEEE), 170–175. doi:10.1109/ICARA55094.2022.9738575

Wang, H., Peng, M.-j., Liu, Y.-k., Liu, S.-w., Xu, R.-y., and Saeed, H. (2020). Remaining useful life prediction techniques of electric valves for nuclear power plants with convolution kernel and lstm. *Science and Technology of Nuclear Installations* 2020

Wehe, D., Lee, J., Martin, W., Mann, R., Hamel, W., and Tulenko, J. (1989). 10. intelligent robotics and remote systems for the nuclear industry. *Nuclear Engineering and Design* 113, 259–267. doi:https://doi.org/10.1016/0029-5493(89)90077-0

Welburn, E., Wright, T., Marsh, C., Lim, S., Gupta, A., Crowther, W., et al. (2019). A mixed reality approach to robotic inspection of remote environments. In *UK-RAS19 Conference: "Embedded Intelligence: Enabling & Supporting RAS Technologies" Proceedings*. 72–74. doi:10.31256/UKRAS19.19

Williams, R. A., Jia, X., Ikin, P., and Knight, D. (2011). Use of multiscale particle simulations in the design of nuclear plant decommissioning. *Particuology* 9, 358–364. doi:https://doi.org/10.1016/j.partic.2010.10.003. Multiscale Modeling and Simulation of Complex Particulate Systems

Wisth, D., Camurri, M., and Fallon, M. (2019). Robust legged robot state estimation using factor graph optimization. *IEEE Robotics and Automation Letters* 4, 4507–4514. doi:10.1109/LRA.2019.2933768

Wright, T., West, A., Licata, M., Hawes, N., and Lennox, B. (2021). Simulating ionising radiation in gazebo for robotic nuclear inspection challenges. *Robotics* 10, 86. doi:10.3390/robotics10030086

Zhao, Y., Diao, X., Huang, J., and Smidts, C. (2019). Automated identification of causal relationships in nuclear power plant event reports. *Nuclear Technology* 205, 1021–1034