Surgical Robotics Environment for NVIDIA Isaac Sim

EN.601.456.01.SP24 Computer Integrated Surgery II

#### Team #8 Final Report

Tae Wan Kim (tkim104@jhu.edu)

April 28th, 2024

# **Table of Contents**

[**Team #8 Final Report 0**](#_95jtzw3bgnlj)

[**Table of Contents 1**](#_ed5172j88eea)

[**1: Overview 2**](#_lfdn0rqv2g5)

[**2: Introduction 2**](#_432ud495kvou)

[2.1: Background 2](#_nrs863cq4bum)

[2.2: Significance 3](#_7z7tgze61m87)

[2.3: Prior Work 4](#_cm5kv98ypa55)

[2.4: Problems / Goals 5](#_l6fq22761f40)

[**3: Technical Summary 6**](#_mguq09d5nvmc)

[3.1: Approach 6](#_hxkglk98nwvd)

[3.2: Dependencies 7](#_s8csc92nnxx8)

[3.3: Timelines 8](#_s65h0bqq6xfh)

[**4: Results 9**](#_k5allnywy7r)

[4.1: Planned Deliverable 9](#_i5f1h3ksyzpl)

[4.2: Results 11](#_ftxk5hth3tbp)

[4.3: Significance 14](#_spur9us6odia)

[**5: Conclusion 14**](#_155ukrw75dol)

[5.1: Reflection 14](#_wb1fqcjthuni)

[5.2: Next Steps 15](#_xh0yg2qeqkeh)

[**6: Management Summary 16**](#_qcdozyo8plhp)

[6.1: Team Members 16](#_ifkj9zr6mit3)

[6.2: Communication 16](#_nqhznxhn0dwx)

[**7: Reading List 17**](#_pc1iy9sebgyf)

[**8: References 18**](#_lz1gsis5ns8)

[**9: Appendix 19**](#_theqemou0m5q)

# **1: Overview**

The Surgical Robotics Environment for NVIDIA Isaac Sim aimed to determine whether or not Isaac Sim, a robotics simulator made by NVIDIA, should be used to model surgical environments. The goal was to determine how well the program could replicate important features one would need to consider when choosing one simulator over another. To do this, required gaining expertise about the Isaac Sim, its functions, and its capacity to serve as a standard robotics simulator. It is also required to check its ability to represent the Surgical Challenge 2021-2022 / Surgical Challenge 2023-2024 [8] outlined in the seminal paper on the Asynchronous Multibody Framework (AMBF) [3], the application used for virtual testing at Johns Hopkins University. Ultimately this project developed the workflow in using Isaac Sim alongside the Surgical Challenge and developed plans and guidelines to further test its use.

# **2: Introduction**

## 2.1: Background

|  |
| --- |

*Figure 1. Omniverse Isaac Sim GUI [6]*

Robotic simulators are used in a wide range of fields from government applications in motion to manipulators used for industry [1]. There is a great amount of literature on the field as a whole, however our project focus targets medical robotics used in surgical applications. Currently, research and development in accurate surgical robotics simulators provide a platform for users to experiment with design and testing before physically creating a product. This primarily reduces the cost and time involved in developing a new surgical robotic device that creators can test without the need to use live testing as much. Relying on the precision and models developed on these simulator platforms, developers have used simulators to further develop the tools to help surgeons or medical practitioners in their work. By reducing the possibility of tremors, and surgery fatigue, and alleviating burdens to the device's user, surgical robotics have become a useful tool to drive better patient outcomes in medicine [2].

Medical or surgical robotics research is a sub-discipline in overall robotics research, thus requiring simulators to have certain features. Most commonly, if a robot is to be teleoperated by a surgeon on a patient, simulators used to develop it would require excellent soft-body calculations, real-time reply, and force/torque feedback for haptic control [1]. Thus, the simulator would then need to model and mimic the real environment for proper development and, more recently, synthetic data creation for training models with machine learning, neural networks, and reinforcement learning to name a few.

*Asynchronous Multi-Body Framework AMBF:*

Developed in 2019, AMBF has been an established surgical robotics simulator with the ability to define the simulation space in its custom AMBF Description Format (ADF). This is a specialized format to house information to launch and run AMBF. The simulator can conduct soft body calculations, supports teleoperation and haptic feedback, and utilizes the Robot Operating System (ROS) as middleware communication [3].

*NVIDIA Isaac Sim:*

Originally launched in 2021 in its Beta, Isaac Sim is a more general robotics environment simulator used in a wide range of applications not just focused on the surgical sphere. Currently, it is on version 2023.1.1 and allows the user to simulate a multitude of features on its PhysX 5 physics engine and portray high fidelity, highly realistic images with its Omniverse RTX renderer [4]. Utilizing the Universal Scene Descriptor file format, Isaac Sim also has full dynamic modeling and supports connections using the Robot Operating System (ROS) as well.

## 2.2: Significance

Over the past few years, the use of machine learning algorithms have exploded in popularity as several disciplines have begun to embrace the technology to solve what once were extremely difficult problems. At the same time, machine learning applications to automate tasks within the surgical setting have increased over time. To train such models on real world machines is expensive and takes an immense amount of time and labeled data. Creating labels manually is a time-consuming but necessary process to train these algorithms, however the use of simulators can automate this process with its synthetic data generation. At the same, these virtual settings allow for easier access to training environments for future surgeons.

Simulators also provide a platform for users to control a model with a real world control and practice with before learning on physical setups. Given the high accuracy of a simulator, the use of virtual training environments allows those practicing to not only have more opportunities to hone their craft, but also additional feedback and data on their performance. Practicing on a simulator with adequate haptic feedback, the trainee could then attain higher proficiency on a certain task like suturing before working on a real phantom or patient.

One final application of simulators goes into its core design philosophy. Simulators mimic the physics of the real world, thus it can act as an excellent testbed for the models and machines themselves. One could answer questions about machine specifications, and performance before spending resources on a physical prototype. There, designers could fine tune, revamp, or completely overhaul several design aspects without the additional costs of recreating the robots themselves. Overall, surgical robotics simulators and simulations as a whole have increased the efficacy of design, training, and machine learning opportunities to be worth further development in the future.

## 2.3: Prior Work

Several papers and research studies have already been done with AMBF, more recently is the its use seen with the Surgical Robotics Challenge:

|  |  |
| --- | --- |

*Figure 2. Surgical Robotics Challenge in AMBF [8]*

The platform has also been used in reinforcement learning and has seen continued use in the surgical simulator space with several projects developed with AMBF [5]. Specifically regarding the seminal paper on AMBF [3], the work done outlines the challenges and key concepts one would need to be aware of when designing and choosing a simulator for a specific task. In the case of surgical robotics, they paid extra care to provide a platform that could handle soft body calculations for skin and muscle. On top of this, they highlight the calculations used and setup of the Surgical Challenge to highlight what surgical robotics require from the simulator to work properly. A specific case is the use of closed loop kinematics, where many machines in this domain often contain "redundant" joints. Additional considerations of articulation and especially collision between bodies require accurate but fast solvers to portray the insertion of a needle or movement of a thread to mimic such a task. The challenge requests accessibility of many of the important features needed to determine whether a simulator is appropriate for surgical environments, ranging from the physics needed to the actual connections and system support for the unique designs in surgical robotics.

On the other hand, multiple industries have already begun using Isaac Sim and have seen research into its development. One group from the University of Toronto has developed a unified simulation framework using Isaac Sim known as ORBIT, which serves as another important reading on the use of Isaac Sim moving forward [4].

## 2.4: Problems / Goals

Broadly, the primary goal is to understand how Isaac Sim could be used as a robotics simulator in the context of a surgical robotics environment, finding out what features it can do well and what it may lack. In more specifics, this goal can be broken down to three sub-goals to achieve for this project in gaining expertise, understanding its performance, and where it could be used.

*Sub-Goal Success Requirements:*

* Gaining Expertise

This sub goal focuses on increasing the literacy we have on the software as previous work on the program has not been conducted for the specific lab. Thus, we require study of the prior works in this field, but also a deep understanding of the documentation to fully comprehend the use of the simulator. At the same time, it is expected to recreate the Surgical Challenge 2021-2022 / 2023-2024 inside the simulator as an end goal to show the use of Isaac Sim. At the same time, a quick start guide of the simulator is necessary to bring others up to speed on Isaac Sim for future work.

* Performance Analysis

This sub goal aims to quantitatively and qualitatively compare the performance of the simulator against AMBF, understanding both simulators' prospects as a useful tool in robotic design in surgery. This step would require formulated metrics to compare against each other to accurately assess both.

* Application

The third goal is highly dependent and mutable depending on the results of the first and second goals that would need to be addressed before accurately determining what success would look like for the Application goal. However, it was determined that the focus would be in generating synthetic data for the use and comparison of ground truths. Ultimately, this would lead into a Sim2Real transfer of a needle or other tool as well. Specifically, this would entail developing an algorithm to conduct segmentation on a needle and thread to determine its estimated pose in 6 degrees of freedom.

# **3: Technical Summary**

## 3.1: Approach

In short, the task is to find out how well Isaac Sim can create a surgical robotics environment. Several steps include learning about Isaac Sim's ability as a robotics simulator, how it measures against other simulators, and how robust it is for research purposes in areas such as segmentation, connectivity, training, and machine learning.

One of the primary steps at hand requires a thorough collection of expertise on NVIDIA's Isaac Sim application to allow users beyond the scope of the project to utilize its libraries, systems, and applications for further research in the lab for surgical robotics environments. While the simulator is already known to provide an excellent platform for robotic simulations in industry, seeing use cases in places like Amazon™ , one avenue we wish to further develop is its use in medical applications. With this comes an emphasis on other aspects of simulation not as considered in different disciplines, such as its ability to perform soft-body calculations.

Ultimately, our approach requires time with Isaac Sim to thoroughly understand what it is capable of and how that can translate to the recreation of the Surgical Challenge. Thus, the initial steps required

| *A. Munawar "A Real-Time Dynamic Simulator and an Associated Front-End Representation Format for Simulating Complex Robots and Environments,"[3]* | *Nvidia Isaac Sim Arm, Nvidia Omniverse [6]* |
| --- | --- |

*Figure 3. AMBF and Isaac Sim Arms*

In terms of measuring it against other simulators, this step involves comparing Isaac Sim against a simulator based on AMBF (Asynchronous Multibody Framework) which was designed primarily for surgical applications for simulating the dynamics of a kinematically redundant robot.

Finally, we hope to see Isaac Sim's use in an actual surgical robotics setting with concepts such as tool segmentation, real-time feedback, and / or machine learning / reinforcement learning algorithms in a surgical procedure such as suturing a phantom. This mainly involves generation of synthetic data that can be compared to known ground truths. Synthetic data is often made based on the visual accuracy and recordings of the simulator, producing depth maps or point clouds that act as the output of the simulation for several other application domains like machine learning and surgeon training.

## 3.2: Dependencies

| **Dependencies** | **Purpose** | **Status** | **Hard Deadline** | **Curr. Priority** | **Alternatives** |
| --- | --- | --- | --- | --- | --- |
| *Desktop System* | To run Isaac Sim and conduct exploratory work | Complete | 2/26 | ↑↑↑↑↑ | Robotorium Computers  Cloud Containers  Mentor's Computer |
| *Robotorium Access* | To use powerful enough desktop for Isaac Sim | Complete | 2/19 | ↑ | Remote Access to desktop  Cloud Access |
| *Robotorium Comp. Account* | To use Robotorium Computers | Complete | 2/19 | ↑ | Personal Account  Mentor's Account |
| *Isaac Sim License* | Access to NVIDIA Omniverse and Isaac Sim environment | Complete | 2/26 | ↑↑↑ | Isaac Sim Enterprise  Isaac Sim Cloud |
| *AMBF-Expertise* | Expertise to run and gather AMBF statistics to compare | Complete | 3/20 | ↔ | AMBF Wiki 2.0 and github  Mentor Expertise |
| *3D Assets* | Surgical Challenge Assets and Files | Complete | 3/2 | ↑↑ | USD File Recreation  Blender Assets |

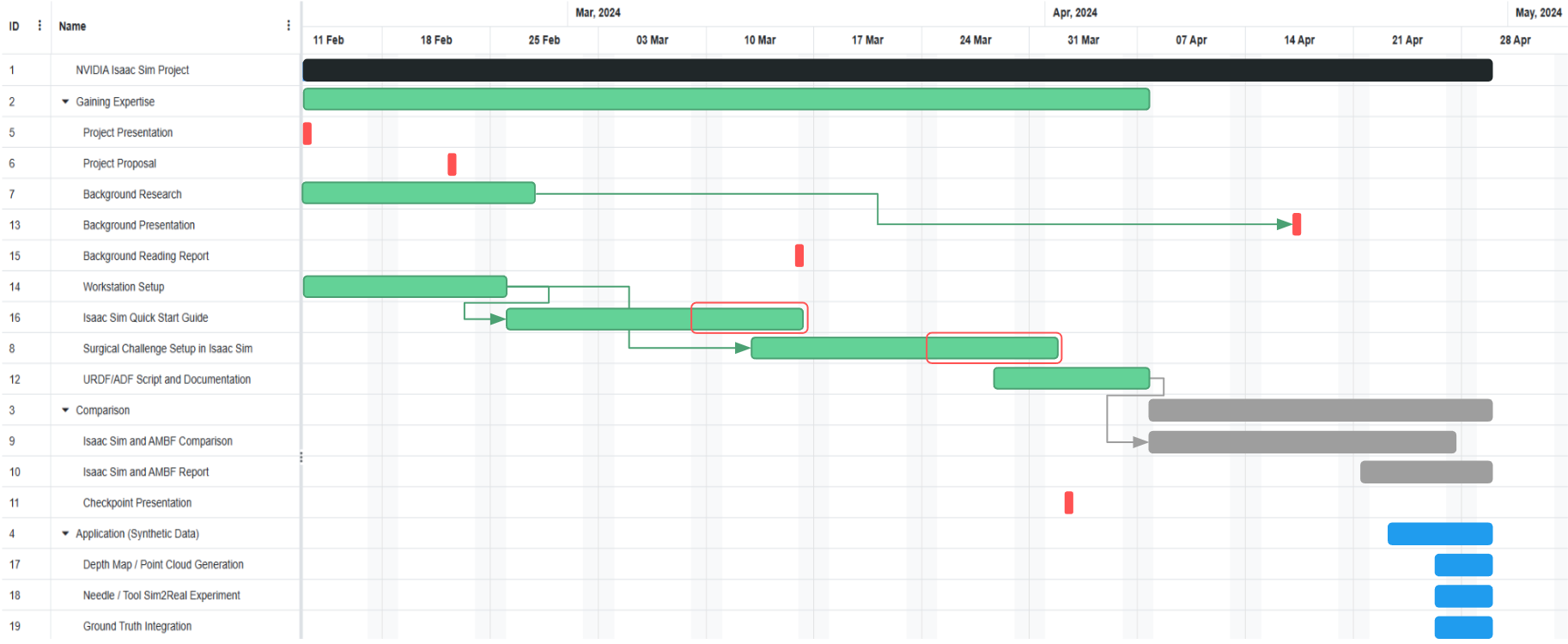
*Figure 4. Required Dependencies of the project]*

An absolutely critical dependency is a workstation or desktop system that can run Isaac Sim. While many computers are available to run other simulators such as AMBF, supported on Linux and Apple software, Issac Sim requires Linux or windows systems and a GPU. At the same time, it is recommended to follow the hardware specifications listed in NVIDIA's website to ensure proper function of the software in development and testing.

Other notable dependencies include access to robotorium computers and the lab itself to allow users to both access the daVinci robot hardware available in the lab, and attain access to machines with ROS and MTM connections for further testing and control.

## 3.3: Timelines





*Figure 5. Previous and Updated Timelines*

From the initial timeline made at the beginning of the project to an updated version, several changes were made to reflect changes made and setbacks that occurred during this project. Specific delays include two weeks where the desktop that ran Isaac Sim encountered fatal issues that prevented progress and required the help of Anton Deguet to aid in debugging the computer itself. This happened in two separate occasions where either an improper install caused CUDA, a toolkit specific to utilizing NVIDIA's gpu, to malfunction, and an update to the driver and systems as a whole that prevented work until a full restart.

# **4: Results**

## 4.1: Planned Deliverable

|  | Deliverables | Deliverable Type |
| --- | --- | --- |
| Minimum | Isaac Sim Quick Start Guide for future usage (installation for JHU systems, execution, and setup) |  |
| Replication of Surgical Robotics Challenge 2021-2022 / 2023-2024 seen in AMBF |  |
| Code and Documentation for Universal Scene Descriptor (USD), Unified Robotics Description Format (URDF), and AMBF Description Format (ADF) possible converters |  |
| Expected | AMBF Analysis / Technical Summaries (While expected to be written by student, still will require mentor expertise) |  |
| Performance Report between Isaac Sim and AMBF |  |
| Demonstration of Isaac Sim and AMBF similarities and differences |  |
| Maximum | Synthetic Code Generation |  |
| ROS / ROS2 connection between Isaac Sim and the da Vinci Research Kit (dVRK) |  |
| Sim2Real transfer with tool/needle segmentation model |  |

*Figure 6. General Dependencies Table with Deliverable Type (Document, Code, Video)*

*Minimum*

The planned minimum deliverables aimed to develop expertise with Isaac Sim and create a basic workflow to allow future users to easily use the application. Rather than poring through NVIDIA's Isaac Sim documentation, a quick start guide was suggested to allow users to familiarize the process of installing and setting up their first scene. Furthermore, to ensure success of this step, a replication of the Surgical Robotics Challenge 2021-2022 / 2023-2024 outlined in the AMBF paper in Isaac Sim was planned. Finally, what was an expected deliverable was shifted down to the minimum in creating a converter to go between a representation of a robot in AMBF's file type (ADF) to an URDF file. As URDF files are more standard and universally used across many platforms, especially systems that use ROS as middleware, it necessitated an easy way to switch between file archetypes to better prepare the groundwork in comparing simulators.

*Expected*

It was expected to essentially conduct several experiments in comparing the capabilities of Isaac Sim to AMBF. This would range between analyzing their performance on physics calculations, visual fidelity, and overall user experience with the applications. Another metric we considered was deployability. While AMBF does not require a GPU to run its computations on, Isaac Sim does.

In general, the expected deliverables included several reports and a comparison video to showcase the differences between the software. These reports would contain information on each program and cover the following:

* + Minimum Hardware Specs ( Deployability )
  + FPS Throughput
  + Realism fidelity ( Likert Scale )
  + Soft-Body calculation Speeds
  + Rendering Speeds

Further information can be found on the Technical Approach Paper Ver 1.0.

*Maximum*

The maximum deliverables highly depended on the results of the first deliverables, mainly understanding what Isaac Sim can and can not do. After spending time in gaining literacy in the program it was decided to narrow down the maximum deliverables to conduct synthetic data generation in Isaac Sim.

Currently, one of the main uses for simulators is to provide a platform to create and record virtual data on the environment designed in it. Thus, it is able to provide a large amount of information and an easy access data set to train machine learning models before deploying and testing such models in a real world setting. For example, AMBF is able to create depth maps on a shown scene that can feed into other applications. Given our expectations on NVIDIA's history on graphical prowess, the main hope is to utilize their visualization pipeline to acquire highly accurate data on such scenes and compare them to ground truths as well. This would have led to developing a Sim2Real transfer with tool segmentation using Isaac Sim as described before.

## 4.2: Results

|  |
| --- |

*Figure 7. Surgical Challenge 2021-2022 in Isaac Sim*

*Quick Start Guide*

Currently, we have been able to write a basic quick start guide on installing NVIDIA Isaac Sim on windows and linux systems. The guide covers basic concepts on the GUI and good to know keyboard and mouse shortcuts to help users navigate the space faster than the documentation available. While the quick start guide is present to rapidly introduce Isaac Sim, it does not serve as a replacement for the actual documentations as they contain greater detail and descriptions of what the simulator is capable of. The guide and an introductory video can be found on the course website.

*Surgical Challenge Scene*

The first step in recreating the Surgical Challenge involved loading in the models into a scene within Isaac Sim. While we aimed to develop a converter between ADF files (AMBF Description Format) and URDF file types (Unified Robot Description Format), a method was developed to utilize the software native converters between several applications.

|  |
| --- |

*Figure 8. Relationship Diagram of Several File Types*

Files that originated from AMBF could be loaded in Blender, a 3D modeling software, with the use of a AMBF Addon to Blender that was made for the Surgical Challenge setup and general ease to create models through visual creation rather than writing out the specific object lines in code. This file can then be exported and imported from Blender into the USD format (Universal Scene Descriptor) that allows both Isaac Sim and Blender to read. At the same time, Isaac has the capabilities to natively convert any USD scene within it to an URDF file without the need for additional extensions or modifications to the existing program.

|  |  |
| --- | --- |

*Figure 9. Phantom Assets in Isaac Sim (2021-2022 Left, 2023-2024 Right)*

Thus, we were able to load in the assets found in the Surgical Challenge 2021-2022 / 2023-2024, mainly the phantoms, daVinci PSM (Patient Side Manipulators), and the needle and thread into the Isaac Sim environment. Immediately, one can make the comparison between AMBF and Isaac Sim on the visual fidelity. While AMBF is able to load the assets as well, the images generated by Isaac Sim are much more detailed, with better focus on the texture of the phantoms. Comparing the rendering quality between figure 9 and figure 2, it is clear that Isaac Sim provides greater detail and realism in its image. Qualitatively, one can argue for the better visual realism in Isaac Sim compared to AMBF.

|  |
| --- |

*Figure 10. dVRK PSM in Isaac Sim*

Finally, we have also loaded in and recreated the kinematic model of the daVinci PSMs used in the challenge into Isaac Sim. The model is articulated to allow the direct control of its movements through the GUI and internal controllers to move each joint. At the time of writing, the model is designed with 8 joints of varying types and captures its collision through a combination of convex hulls and meshes to accurately portray its rigid body physics. Looking into how it is articulated, the model also utilizes a series of drivers for each joint with position based values to control how it moves.

*ROS1 Connection*

|  |
| --- |

*Figure 11. ROS Control Blocks in Isaac Sim*

Jumping ahead to a part of the maximum deliverable that was suggested earlier into the project, we then worked to incorporate ROS1 into the simulator, allowing one to communicate through ROS to control the model inside Isaac Sim. Utilizing Isaac Sim's own visual scripting capabilities with control nodes and graphs, we were able to easily connect to ROS and generate custom ROS topics and headers to transfer information from one machine to another. Based on the Collaborative Robotics Toolkit (CRTK) we can customize the name of the headers to match this convention for standardized communication. As of the time of writing, we have completed the full pipeline / connectivity to go between Isaac Sim and a physical Master Tool Manipulator (MTM) through ROS to control the model with a real world controller. However, more development is needed to increase the accuracy of control as initial tests indicate gross errors between the orientation of the model and MTM. More testing and debugging is required.

## 4.3: Significance

While we were unable to reach completion of the expected and maximum deliverables, the work and results so far has built the foundations for future work on the project. With the quick start guide of Isaac Sim and the standing example of the Surgical Challenge scene loaded into Isaac Sim, we have the basic platform to show Isaac Sim's potential to develop surgical environments for testing and training of algorithms and surgeons. The initial preparations for testing appear to be almost finished, thus with correction to some code and workflows, the project has opened the gates to move into the experimentation steps mentioned in the expected and maximum. Looking ahead, more effort is now needed to test the systems, the workflows, and the software through performance analysis and application.

# **5: Conclusion**

## 5.1: Reflection

Overall, the project was an eye-opening experience to learn and develop the workflow in using Isaac Sim to remake the Surgical Challenge 2021-2022 / 2023-2024. A great deal of information needed to be synthesized to understand the process as a whole and to make progress during the semester. While there were many challenges that arose during the course of the project for the Spring 2024 semester, I believe we were able to make the best of the situation and still achieve decent progress in total.

*Challenges*

The main challenge for this project was the great dependency on a usable workstation to run NVIDIA Isaac Sim and meet all system requirements to also run several other applications. At the beginning of the project, the bottleneck for progress was access to a computer that had a GPU to handle the visual demands of the simulator in question. While an alternative workstation was available it was not until a few weeks into the semester a dedicated desktop was purchased and used for the project. At the same time, some work still required the shared station to connect to a MTM, ROS1 stack, and Isaac Sim at the same time to compare against AMBF. Unfortunately, several times throughout the timeline, the computer had severe errors that prevented any work on them that slowed progress.

*Lessons*

Nonetheless, the experience learned throughout this was fascinating and invaluable to anyone who wishes to continue their study in computer aided surgery. As the use of virtual environments to train novice surgeons and machine learning algorithms has been gaining traction in the past few years, it is a growing field that will see development in the future. Not only did I have the opportunity to learn about Isaac Sim itself, how the simulator worked, but also how simulations work in general. On top of that, I had many lessons on AMBF, its file structures and ways to represent robots in an efficient manner. Many programs beyond simulators, have certain ways to represent a scene or body which highlights what type of information is important for that program. Some software focus on the visual representation, while others place concern on the way bodies are connected and what type of joints are allowed. Closed kinematic chain / parallel linkage robots are commonly seen in medical robotics, yet not every simulator allows such type of connection.

Other topics include kinematic chains or articulation roots, the fundamental building blocks to begin controlling a collection of bodies in a systematic way. At the same time, it has been a great opportunity to learn about ROS and the CRTK format in a standardized way to communicate between machines. As a middleware, ROS appears to be the accepted method to connect between the many facets of surgical robotics.

## 5.2: Next Steps

Moving forward, the project will continue beyond the semester in completing the workflow to compare Isaac Sim and AMBF. Immediately, the focus will involve correcting the teleoperation control of a dVRK PSM model inside Isaac Sim with a MTM, specifically the controller found in the Laboratory for Computational Sensing and Robotics.

Several physics and visualizations tests are still required to fully understand the differences and similarities between Isaac Sim and AMBF. At the same time, as several developments are made to incorporate Isaac Sim's architecture with the available tools at Johns Hopkins are made, further experimentation with the simulator can be conducted. Depending on future progress, the hope is to utilize Isaac Sim to generate synthetic data, in the representation of point clouds or depth maps, to then compare with existing ground truths. Furthermore, the ultimate goal outlined in the maximum deliverable was to set up and use a Sim2Real transfer of the Surgical Challenge, and especially segmentation of the tool used in that environment.

# **6: Management Summary**

## 6.1: Team Members

*Student Members:*

* **Tae Wan Kim** (tkim104@jhu.edu)

Is primarily responsible for bringing the NVIDIA Isaac Sim project and applications to a usable state for the lab

*Mentors:*

* **Dr. Peter Kazanzides** (pkaz@jhu.edu)
* **Dr. Adnan Munawar** (amunawa2@jh.edu)
* **Hisashi Ishida** (hishida3@jhu.edu)
* **Juan Antonio Barragan** (jbarrag3@jhu.edu)
* **Jintan Zhang** (jzhan247@jhu.edu)

## 6.2: Communication

*Meetings:*

There was a joint meeting between Team # 8: Surgical Robotics Environment for NVIDIA Isaac Sim and Team # 12 Digital Playground for developing medical robotics environment that occurs every Friday, 2:00 - 3:00 pm. At the same time, the mentors have provided schedulable meeting times throughout the week if necessary.

During the second half of the semester, the meeting time was changed to every Friday, 12:30 - 1:30 pm to accommodate schedules conflicts for the student for classes.

*Platforms:*

* Communication
  + Microsoft Teams: Primary way of communication is a Microsoft Teams channel between the student and mentors.
  + Email and Outlook: Additional way for communication between student and mentors, used as needed if Microsoft Teams is insufficient.
* Documentation
  + Microsoft Teams: File sharing done in the teams channel for mentors to review student produced work.
  + Google Docs: Student documentation, drafts, and notes in private google docs account, shared through Teams for mentors.
  + CIS2 Course Wiki: Project Wiki hosted on the CIS2 website, central place to store files and documents that are related to the project

# **7: Reading List**

**Quiroz Omaña**, J. J., Marques Marinho, M., & Harada, K. (2023). *Digital Twin of a Multi-Arm Robot Platform based on Isaac Sim for Synthetic Data Generation* (Version 1.0.0) [Data set]. Zenodo. <https://doi.org/10.5281/ZENODO.7860757>

**Mittal, M**., Yu C., Liu, J., Rudin, N., Hoeller, D., Yuan, J., Tehrani, P., Singh, R., Guo, Y., Mazhar, H., Mandlekar, A., Babich, B., State, G., Hutter, M., & Garg, A. (2023). *ORBIT: A Unified Simulation Framework for Interactive Robot Learning Environments.* [arXiv:2301.04195](https://arxiv.org/abs/2301.04195) [cs.RO]. <https://doi.org/10.48550/arXiv.2301.04195>

**Schmidgall, S**., Krieger, A., & Eshraghian, J. (2024). *Surgical Gym: A high-performance GPU-based platform for reinforcement learning with surgical robots.* [arXiv:2310.04676](https://arxiv.org/abs/2310.04676) [cs.RO]. <https://doi.org/10.48550/arXiv.2310.04676>

**V. M. Varier,** D. K. Rajamani, F. Tavakkolmoghaddam, A. Munawar and G. S. Fischer, *"AMBF-RL: A real-time simulation based Reinforcement Learning toolkit for Medical Robotics,"* *2022 International Symposium on Medical Robotics (ISMR)*, GA, USA, 2022, pp. 1-8, doi: 10.1109/ISMR48347.2022.9807609.

**A. Munawar**, J. Y. Wu, G. S. Fischer, R. H. Taylor and P. Kazanzides, "Open Simulation Environment for Learning and Practice of Robot-Assisted Surgical Suturing," in *IEEE Robotics and Automation Letters*, vol. 7, no. 2, pp. 3843-3850, April 2022, doi: 10.1109/LRA.2022.3146900.

**Zhou, Zhehua** & Song, Jiayang & Xie, Xuan & Shu, Zhan & Ma, Lei & Liu, Dikai & Yin, Jianxiong & See, Simon. (2023). Towards Building AI-CPS with NVIDIA Isaac Sim: An Industrial Benchmark and Case Study for Robotics Manipulation.

**J. Collins**, S. Chand, A. Vanderkop and D. Howard, "A Review of Physics Simulators for Robotic Applications," in IEEE Access, vol. 9, pp. 51416-51431, 2021, doi: 10.1109/ACCESS.2021.3068769.

**Muratore F**, Ramos F, Turk G, Yu W, Gienger M, Peters J. Robot Learning From Randomized Simulations: A Review. Front Robot AI. 2022 Apr 11;9:799893. doi: 10.3389/front.2022.799893. PMID: 35494543; PMCID: PMC9038844.

# **8: References**

[1] **J. Collins**, S. Chand, A. Vanderkop and D. Howard, "A Review of Physics Simulators for Robotic Applications," in IEEE Access, vol. 9, pp. 51416-51431, 2021, doi: 10.1109/ACCESS.2021.3068769.

[2] **Schmidgall, S**., Krieger, A., & Eshraghian, J. (2024). *Surgical Gym: A high-performance GPU-based platform for reinforcement learning with surgical robots.* [arXiv:2310.04676](https://arxiv.org/abs/2310.04676) [cs.RO]. <https://doi.org/10.48550/arXiv.2310.04676>

[3] **A. Munawar,** Y. Wang, R. Gondokaryono and G. S. Fischer, "A Real-Time Dynamic Simulator and an Associated Front-End Representation Format for Simulating Complex Robots and Environments," 2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Macau, China, 2019, pp. 1875-1882, doi: 10.1109/IROS40897.2019.8968568.

[4] **Mittal, M**., Yu C., Liu, J., Rudin, N., Hoeller, D., Yuan, J., Tehrani, P., Singh, R., Guo, Y., Mazhar, H., Mandlekar, A., Babich, B., State, G., Hutter, M., & Garg, A. (2023). *ORBIT: A Unified Simulation Framework for Interactive Robot Learning Environments.* [arXiv:2301.04195](https://arxiv.org/abs/2301.04195) [cs.RO]. <https://doi.org/10.48550/arXiv.2301.04195>

[5] **V. M. Varier,** D. K. Rajamani, F. Tavakkolmoghaddam, A. Munawar and G. S. Fischer, *"AMBF-RL: A real-time simulation based Reinforcement Learning toolkit for Medical Robotics,"* *2022 International Symposium on Medical Robotics (ISMR)*, GA, USA, 2022, pp. 1-8, doi: 10.1109/ISMR48347.2022.9807609.

[6] **Omniverse, NVIDIA**. *Isaac Sim Introduction*. What Is Isaac Sim? - Omniverse IsaacSim latest documentation. https://docs.omniverse.nvidia.com/isaacsim/latest/index.html

[7] **A. Munawar**, J. Y. Wu, G. S. Fischer, R. H. Taylor and P. Kazanzides, "Open Simulation Environment for Learning and Practice of Robot-Assisted Surgical Suturing," in *IEEE Robotics and Automation Letters*, vol. 7, no. 2, pp. 3843-3850, April 2022, doi: 10.1109/LRA.2022.3146900.

# **9: Appendix**

*Guides / Quick Start / Reports*

The quick start guide and further descriptions of Isaac Sim can be found in the course wiki: [courses:456:2024:projects:456-2024-08:project-08 – CIIS Wiki (jhu.edu)](https://ciis.lcsr.jhu.edu/doku.php?id=courses%3A456%3A2024%3Aprojects%3A456-2024-08%3Aproject-08). Specifically, the manual can be found at the bottom of the page under "Other Resources and Project Files" alongside several reports and documentation of progress logistics such as Technical Approach file and System Specifications.

*Code and Assets*

Code and assets are also available on the course wiki: [courses:456:2024:projects:456-2024-08:project-08 – CIIS Wiki (jhu.edu)](https://ciis.lcsr.jhu.edu/doku.php?id=courses%3A456%3A2024%3Aprojects%3A456-2024-08%3Aproject-08). One can mainly find the assets used in Isaac Sim specifically in the "Other Resources and Project Files" section that will contain links to files used in the quick start and actual scene recreation for Isaac Sim Surgical Challenge.