TELEOPERATION INTERFACE FOR AERIAL TRANSPORTATION SYSTEMS

By

ZHENGHE, CHENLI

MSc RESEARCH PROPOSAL

Department of Engineering
UNIVERSITY OF BRISTOL

1 Introduction

The research expects to develop a customised teleoperation interface for aerial transportation systems using the Unity game engine. The teleoperation interface will combine hardware (input devices) and software to create a human-computer interface (HCI); it will include a 3D virtual environment, simulated sensors and control algorithms to function as a ground control station (GCS) effectively. This interface will allow a user to remotely control a group of UAVs to perform cooperative payload transport missions. The proposal will introduce the aims and the detailed objectives of this project, then elaborate on the project's motivation with supported background research; risks and impacts will also be considered.

2 Aims and Objective

2.1 Aims

This project will use Unity to develop a teleoperation interface and connect it to a multi-drone transport system, running in a separate C++ program, using shared memory or other technology like Robotic Operating System (ROS) through TCP connection. These three subsystems: **the human-computer interface**, **the group of UAVs** and the **link between them** will behave as the local site, the remote site and the communication channel of a teleoperation framework. The project will seek and develop a proper graphical user interface(GUI) system and human-interaction method(like a keyboard and Gamepad) to find suitable ways to control a multi-drone cooperation system, which provides researchers and users with a better intuitive interaction increasing situational awareness. Furthermore, the Unity game engine is responsible for the **human-computer interface**, building a realistic outdoor environment and a complete user interface to provide rich experimental data display. Then the Game Engine will help the researchers build a better interactive experience using a keyboard, joysticks, and even force feedback controllers; the remote control will also be simulated. This research will also cooperate with other ongoing investigations to assist and evaluate the research's algorithms and teleoperation methods[1]. The supervisor will provide an external program; it will simulate realistic dynamic models of the drones, ropes and payload, along with the low-level control on each vehicle (PID-based velocity control). This program will be launched from the teleoperation interface and run without a GUI. Figure 2.1 shows the entire project's expected structure.

2.2 Objectives

1. **Build the interaction system using an efficient and versatile game engine with**

   The project will choose the Unity engine to develop. The interface will act design as a ground control station base on Unity. We will also use unity to build a high-level control system to ensure it is feasible to control the multi-drones simultaneously. Unity will then communicate with the group of UAVs (which will be simulated in a separate C++ program) using shared memory and ROS.

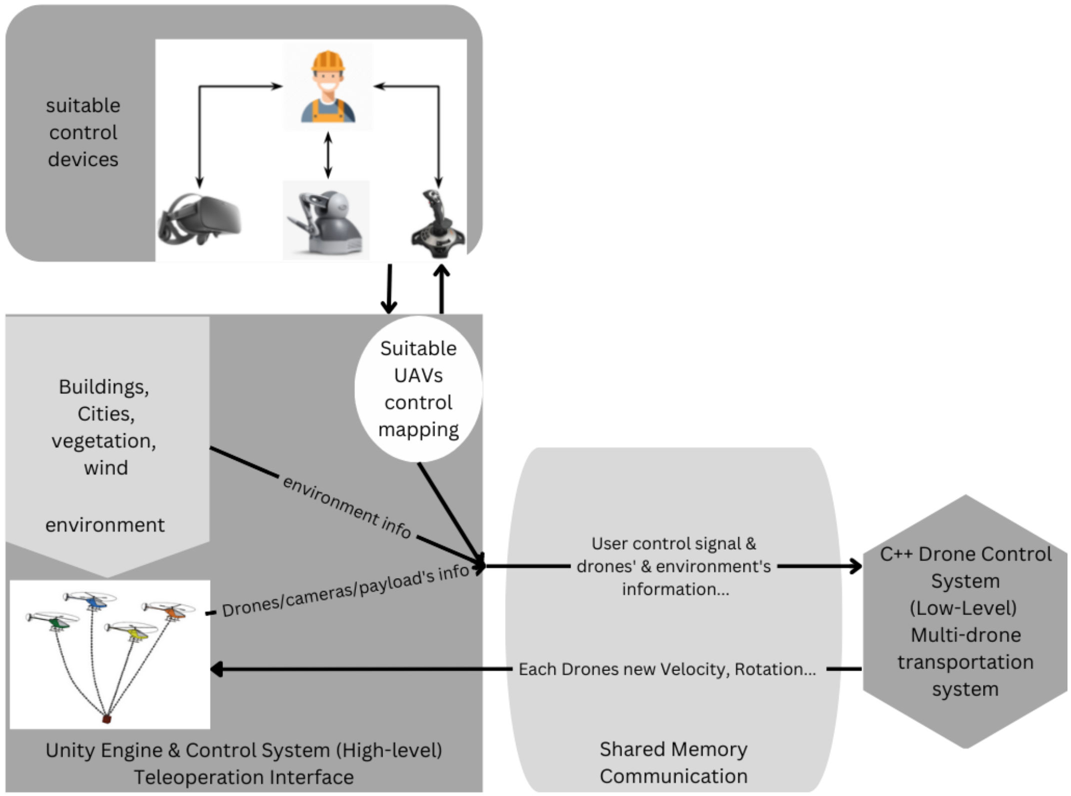


Figure 2.1: Project Structure

2. **Build an outdoor virtual environment for multiple UAV transportation tests**

   Unity will take the environmental information(obstacles, wind, buildings and any obstacles) and then send this information to C++ control programs to let the drones react. These tasks require the flexible finding/use of various components and assets in the game engine. Also need to implement algorithms to simulate wind and other possible environmental variables.

3. **Building an exemplary user interface for detailed data display**

   (a) The user should have intuitive access to all the critical information of the simulation (multi-cameras, altitude, speed, trajectory, position.) and organise it logically on the screens.

   (b) The user interface should be simple and efficient, with windows text laid out neatly and clearly (it would be better to support various languages).

   Unity's UI components will be the primary tool for this goal[2]. However, they will also rely on other established components for implementation and refinement. Packages may need for multi languages localisation.

4. **High-Level Control Algorithms**

   (a) Teleoperation control, where the user input is processed (combining multiple inputs like keyboard, gamepad, haptic device, etc.; filtering, mapping functions) to obtain a correct trajectory/path reference to control the drones' formation.

   (b) Explore and evaluate current formation algorithms[1]. it will need to develop and investigate motion control, load equalisation and vehicle orientation, and the formation algorithms will be adapted, tested and modified as the project progresses.

(c) Develop suitable algorithms, interfaces, and logic to represent the experiment data.

These above require C++, C#, programming, Unity Engine and shared memory knowledge, and they also need more algorithms and robotic control basics.

5. Real-time data sharing

(a) This software may need to read/write using the shared memory or other technology to exchange data between Game Engine software and the drones' control programs(may need to simulate the teleoperate command in real life, such as latency).

Read/write will use shared memory to cooperate independently; changing and modifying the C++ programs to C# in Unity could be the best practical method. Could use Data Distribution Service (DDS) used by Robot Operating System (ROS2) and thus interoperate with the flight control program (C++)[3].

6. Provide a better working experience.

(a) The user should be able to control the multi-UAV system in the virtual 3D environment via the keyboard and mouse.

(b) This research will study the best way to design and map the user's command to actual control inputs in the air transportation system on the remote side. Users should be able to control the aircraft quickly and efficiently and receive meaningful feedback in real-time.

For better control and information acquisition, there will be at least a joystick/Gamepad or force feedback/haptic device (e.g., 3DOF Novint Falcon translational) to control the muti-UAV system. Unity has a wealth of proven device support solutions. However, it will take a lot of research and practice to find the right approach to mapping the control signal with the control program.

7. Portability

(a) The program should be adapted to other similar research projects with superficial modifications.

Unity supports multiple platforms, so more technical requirements will come from good code development, management, and proper project planning[2]. The final version could be a build software or a transferable Unity project file to be considered.

8. Additional features: attempt to implement radar detection via ray or ray tracing; Time-Delay; VR devices can be added for a more immersive 360-degree experience

It would be simpler to implement radar-like effects through the game engine. However, the project expects to implement more realistic lidar scanning principles, requiring more complex technology to implement and simulate radar, LiDAR, and point cloud may be developed. VR devices might give more intuitive feedback on the controls, so implementing VR functionality would complete the project. The project may need and consider time delay situations in the real-life remote control.

3 Motivation

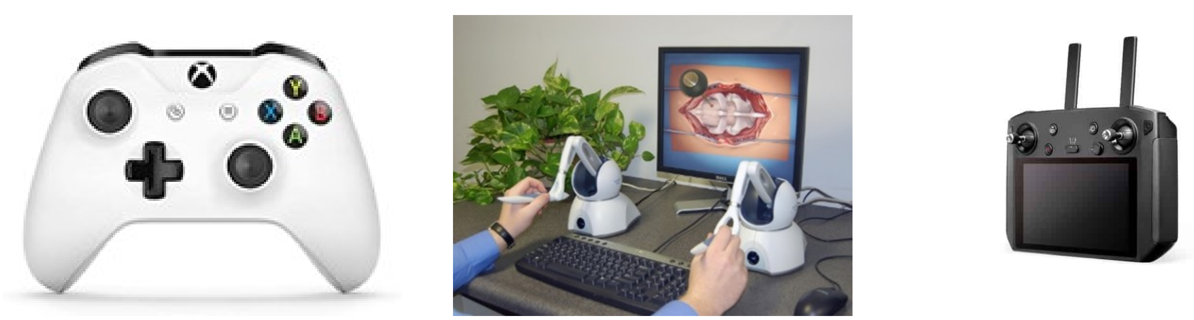
1. **Testing Platform requirement**

   This project has some related research. In [4], the authors developed and simulated a single UAV bilateral teleoperation task with no payload. In[5], an autonomous cooperative payload transport using two UAVs is presented. In [1], a similar control technique is extended and adapted to n number of UAVs transporting the payload in forward flight." This research was based on C++ programs or MATLAB(Simulink). These research methods helped the research to some extent. Still, they found building a good, efficient immersive outdoor 3D environment(City and other ample space) challenging and could not correctly simulate the research's teleoperation tasks based on these platforms. Similar situations happened on other Robotics simulation software like Gazebo, Webots, CoppeliaSim and other popular simulation platforms[6]. In these cases, a better outdoor virtual environment and physics systems are necessary for better algorithm testing.
   In addition, the early project is not compatible with multiple platforms (currently limited to Windows), and there is a great need for more prosperous and flexible tools/interfaces. Research teams may need much time to develop the lacked functions base on traditional simulators. However, most features already have on the game engines, so integration and faster testing methods are urgently needed[7]. Game Engine could be an ideal solution since it has more rich functions and extensibility.

2. **Teleoperation System demand**

   Currently, the ways to control swarm drones are still being explored[8]. This research will find some proper methods to control several drones at once. More control hardware support and highly flexible UI development could help us find possible ways to control the swarm robots. Research on human-computer interaction with drones in other areas is worthy of reference and improvement[9].

   Using cheap, standard hardware is easy to start with the controller. Gamepads are an excellent choice because they are reasonable, familiar, highly explorable and accurate. Other more expensive devices, such as haptic feedback, intelligent controller and aircraft joystick, are also worth considering.



(a) Gamepad (b) haptic devices (c) Smart Controller

Figure 3.1: Some drone control types of equipment

3. Multi-drone teleportation for transportation

Drones have a vast potential for transportation, as they have little space impact and can transport goods more efficiently[10]. Single multi-rotor-drone have difficulty in power and range for specific situations (long distances, high-weight cargo)[11]. Thus, collaborative multi-drone transport has become one essential research area[12] [11][13][14]. But there is still minimal research on teleoperation interfaces for multi-drone cooperative payload transport and simulations, so this project will enrich and explore this domain and give more reliable and proven solutions.

As with swarm control, the multi-drone vehicle is more complex and requires more problems to be overcome[1]. Such as how to maintain the distance between drones; how to work together to ensure that the position and trajectory of the cargo are stable while overcoming obstacles and wind speeds; how to overcome height differences, and how to overcome cargo oscillation. These problems can be alleviated using the cable connection, which is more suitable for cooperative transportation of multiple UAVs[1][12].

Although the focus of this project is not on the development of the algorithm, there is still a need to deploy and optimise the algorithm after developing the test environment, as the previous research in this project was more based on theory and simulation in an ideal state, so new problems will likely arise after adding more environmental variables, so there is still a need to analyse and evaluate the tested algorithm and even improve it.

This project will test teleoperation and formation control algorithms combining previous research[5][1], adapting them to a multi-lift teleoperation framework. These previous studies were more based on theory, and the environment simulation was nearly ideal; in contrast, in this project, new problems will likely arise after adding more environmental features, so the control algorithms will be tested, analysed and improved as the project evolves.

Figure 3.2: Multi-Drone Teleoperation Task showcase[15]

4 Literature Review

UAV-related and game-engine-based projects already exist that simulate realistic environments and mission scenarios to some extent and are worthy of reference and study. [16][17]. Still, unfortunately, they have not yet conducted any or developed simulations for cooperative multi-drone transport or lack a broad outdoor environment, nor have they explored scenarios for cooperative multi-drone control. Hence, this project is still worth implementing.

- **Rotary-wing UAV and Multi-drone System**
  - **(Multi)Rotary-wing drone**

    Drones have become very popular today, with different types of drones helping people in all corners of the world accomplish various tasks in a more environmentally friendly and efficient manner, especially in transportation, military and reconnaissance missions [10]. Rotary-Wing UAVs spend most of their power overcoming gravity, so they usually have very low power efficiency and carry fewer payloads [18]. However, it must be admitted that because rotorcraft can hover, are easy to control and have less space requirement, and they are well suited to low altitudes and challenging terrain with the high precision operation and many activities; besides, the combination of fixed/rotor drone also become mature gradually for both types of drone's features[19].
  - **Multi-drone cooperative transport**

    As mentioned above, because of the outstanding manoeuvrability of rotary wings, drones are expected to play a better role in short and medium-range transport[12]. As a single UAV's range and load capacity are minimal, multi-UAV cooperation (more than two) will likely be the only solution to spread the load well over multiple UAVs[11][1].

    The control algorithms for swarm UAVs have always been a very challenging topic [8]. Solutions include but are not limited to parallel control, distributed control, centralised control, neural network control and multi-solution collaborative control, all of which can control swarms of UAVs to some extent[8]. However, none of them has a perfect and mature solution, and the field of aerial multi-UAV transport is also very scarce. This study attempts to explore a more efficient and stable control scheme under the premise of centralised control[1], with a ground-based station combined with specific communication methods to centrally teleoperated a distant swarm of UAVs, using manual control to intervene in the swarm of UAVs working together, when manual and collaborative control systems mature, autonomous driving will also become possible in the future.

    Coordinated control of multiple vehicles flying with a suspended load is a challenging and dangerous task, as the suspended payload is an oscillating pendulum, so the payload can significantly alter the flight characteristics of the vehicles. The formation structure of such a system is a challenge and the aim of the study, as is the need to control multiple UAVs in concert, ensuring that they do not collide and that cargo and UAVs follow a prescribed route and avoid obstacles. Although several studies have provided reliable multi-drone transport solutions, they have more or less failed to consider the effects of environmental factors (wind and barriers)[4], so this project and its predecessor needed to incorporate ecological elements to investigate more reliable multi-drone control mechanisms better.

- **Teleoperation System**

  A remote-control system for multi-drone transport will not be as simple as a consumer-grade drone operated by remote control; it will need to be able to control drones that are far away and sometimes are in a dangerous area or area that the human is hard to reach, as shown in 4.1. Communication and how to manage and map the state of drones becomes a critical factor in the non-man-machine remote operation[20]. The method of control and communication will be something that needs to be considered in this section.

  Traditional drone control is usually via dedicated Radio Control (RC) remote control devices, which often have a high threshold for operation. Some emerging and simplified drone controllers allow for a lower control threshold, with fewer buttons and the ability to act as a display via mobile devices such as mobile phones in Figure3.1. However, these devices cannot control multiple drones at the beginning of the design. Teleoperation control means the opportunity to control and manage multiple drones from the ground centrally. They are not as straightforward as traditional remote controls but have more significant advantages regarding distance, security and multi-terminal management[20][21]. This project will seek and test some solutions that allow some of the cheaper, more straightforward and more approachable control methods to be used for complex UAV swarm control.

  Remote is the project's final goal, as UAV transport is usually long-range and has specific requirements in terms of accuracy. Commonly used control methods such as Wi-Fi and 9000MHz/2.1GHz/2.4GHz/5.8GHz long-wave signals are insufficient for long-range control. 5G networks are feasible, but the coverage is too small, 4G coverage is much broader, but its latency is too high, so 4G+5G or even the addition of specific technologies may be an option to consider in the future [22][21][20]. This project does not yet necessitate the construction of an actual ground base station. However, it is necessary to consider and simulate the problems associated with remote communication in the control program: latency and packet loss, which are more evident and critical when a large amount of information is to be transmitted [5].

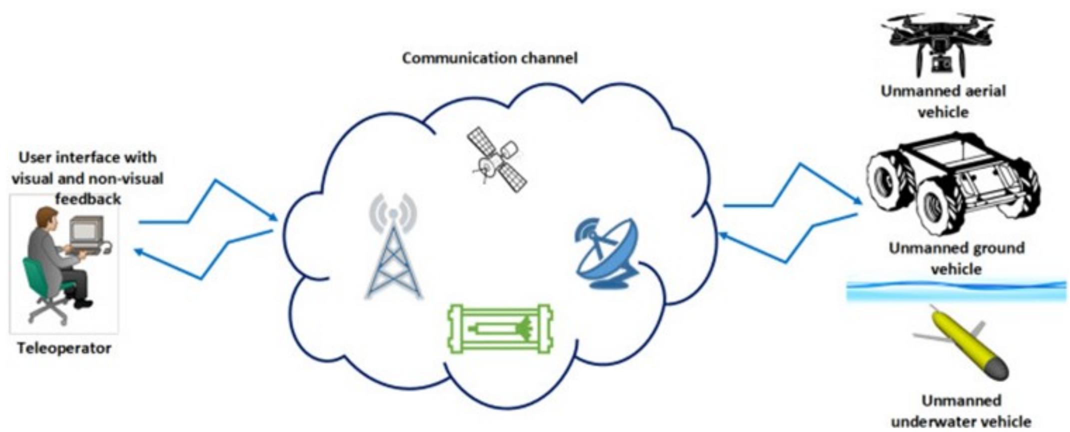


Figure 4.1: Teleoperation structure

- **Traditional robot simulation platform shortage**

  In previous research activities, various robot simulation software such as CoppeliaSim, Gazebo and MATLAB (Simulink) were tried. However, it was found that many of the desired features (e.g., performance, outdoor 3D, multi-platform, simulation scenarios, realism, and interactivity) were obstructed[23][24][25]. There are also many different kinds of robot simulation platforms on the market, all of which have their orientation, so the platforms for different types of robots may be very other [6]. Robot simulators have indeed been well developed. They can be helpful in testing and simulating early robot development processes, saving time and cost, so they have primarily facilitated research and commercial activities[26].

  However, perhaps because they focus more on simulation, they often lack suitable user interfaces and graphical rendering capabilities or the extensibility and single functionality to meet some experimental conditions requiring more comprehensive and complex environments. Some of the newer robot simulators have identified these problems and have attempted to integrate robot simulators with game engines [6]. Another reliable solution is to use game engines directly, as they also have full m-simulation capabilities. Trying to build a targeted simulation engine can be very time-consuming and costly. At the same time, some of the features expected to be developed may already be available in mainstream game engines, so it may be an excellent option to create new technologies directly from existing ones.

- **Game Engine**

  - The gaming industry has evolved over the decades, directly contributing to developing iterations of professional 3D engines - game engines that have become increasingly efficient and high-quality, to the point where they are fully capable of handling robot simulations and research [7]. Most game engines contain the following systems: rendering engines (including 2D and 3D graphics), physics engines, collision detection systems, sound effects, scripting engines, computer animation, artificial intelligence, network engines, and scene management [27]. Game Engines make game development much more efficient. It also makes robotics simulations and other research projects much more effective, allowing developers to focus more on the implementation of the project itself rather than on the underlying functionality. Many research and experimental shreds of evidence show that the game engine for robotics research has been successfully applied [23][28][16][17].

    Like most game engines, Unity has all the features and functionality of a game engine. However, Unity has outstanding advantages, such as solid developer support, the largest community, multi-platform support, assets store and other[2]. So this project uses Unity as the primary tool is feasible.

- **Robot Operating System**

  - Robot Operating System(ROS) is a computer operating system architecture designed specifically for robotics software development. It is an open-source meta-operating system (post-operating system) that provides operating system-like services, including hardware abstraction, underlying driver management, execution of standard functions, inter-programming messaging, package management, and tools and libraries for acquiring, building, writing and executing multi-computer fusion programs[29]. The Unity engine and ROS already have a more mature interoperability solution, which allows Unity to perform robot simulation tasks better[24][3][7]. ROS will likely be used during the implementation phase of this project to aid communication between programs, and it is also likely that some features will be used to refine the realistic behaviour of the drones, so ROS is also a tool worth considering for this scenario

5 Impact Assessment

- **Social/Cultural**

  Firstly, this project is most likely to contribute to multi-drone coordinated transport. There is a huge potential demand for the additional potential that multiple rotary-wing drones can bring[8][5].

  The project is also expected to be useful for people with some disabilities. As the project envisions a rich visual interface and somatosensory control(vibration or force/haptic feedback), people with hearing impairments and disabilities could access essential information through visual or hepatic with more straightforward controls. The project hopes to bring the robotics industry to a broader audience and stimulate the interest of non-researchers. As the project aims to be multi-platform and based on a game engine, it is expected that a wider group of people can use the relevant extensions on a broader range of devices. The development of scientific robotics and kinematics in the gaming industry can significantly increase the realism and fun of the games, which on the one hand, can make game development better and, on the other hand, may also stimulate the interest of particular groups of people in science[7].

- **Technological**

  The project certainly contributes to the drone industry first and foremost, as described in the previous article and the previous section. Moreover, this project is expected to enrich and diversify research activities. The scenarios used are still based on the physical world, and the data is presented almost the same way. So, it can be applied to other robotics projects to some extent. It should also help some researchers to validate their programs and designs better while being richly flexible in terms of scalability and changeability. Research experiments can focus more on the project's pre-development, allowing them to experiment with ideas and solutions in a virtual world without worrying about real-life factors (safety, environmental protection, cost)[26]. Moreover, scientists can quickly conduct more experiments and iterations and process/collect more experimental data. Perhaps this is also helpful for future AI or big data-related research. However, a problem that must be considered is game engines/robotic simulators do not fully simulate real-world factors (such as complex aerodynamics and more specific and rigorous mechanics)[26]. Like the other robotic simulation platforms, they are just a reference to algorithms and fundamental physics; they cannot serve as an absolute reference for the later stages of the experiment. It may be somewhat misleading that researchers must be aware of this.

- **Environmental**

  The project is expected to be very friendly to the environment. This project only needs to be network and computer-based, requiring very little power, network and storage consumption. Hence, the environmental issues are more built into the computer itself[30]. The project largely avoids the problem of having to simulate experiments in the real world by not using many materials and environments to build drones and mission scenarios. The experimenters must try out various systems in the virtual scenario. Hence, the project itself is currently very safe and environmentally friendly. However, the problem becomes more complex in the later stages of the project when multiple drones are deployed. While the drones did reduce CO_2 and other air pollutants to some extent, they caused other problems[31]. The noise, propellers and collisions of drones are likely to cause serious harm to plants, birds and other animals (including humans) and may endanger other aircraft. Parts and cargo on drones can also be dropped, leading to safety accidents and property damage. In addition, drone batteries and other components remain environmental issues[32][31]. However, this project aims to use small drones in concert, and smaller drones are more environmentally friendly than larger ones[1]. These factors are not something that the project needs to consider at the moment but will need to be considered and addressed in the future, primarily when implemented on a large scale.

- **Ethical**

  As the project is currently built in a virtual environment, it is unlikely that real-life data will be collected, so there are no privacy or other people's data security concerns. The only information required for this project is the collection and use of potentially realistic modelling and maps of urban building complexes, as well as feedback that will be collected from multiple test users on their experiences(Gender and disability without names). The real-world map information will be from the publicly permitted resource (Google, Bing Maps). It is worth noting that the follow-up to this project will be in real life. Noise, safety, privacy and other issues have always been ethical for drones[33][31]. Although the project currently will not have practical experiments designed to address these issues, it is still necessary to consider them in future research so that these issues will be properly considered in the project's simulation scenarios for future research.

- Economic

  The overall economic benefits of the project should be very significant. Firstly, it does not require the real-world validation of various scenarios from earlier projects or research developments, so there is less need to purchase a large amount of experimental equipment (e.g., Computers, 3D models, material files, literature, and algorithms). So, there are significant research costs (equipment, maintenance, experimental space) savings. Moreover, researchers can try more solutions more quickly and efficiently based on simulation environments, thus speeding up the overall projects. Project software can be reused and iterated upon, creating more versatile simulation software to facilitate research activities better, thus creating a virtuous circle. In addition, multiple drones working together means numerous, relatively more little drones can be used, and smaller drones tend to be cheaper[1]. At the same time, their spare parts are also more affordable, more readily available, and more flexible for replacement and repair.

- Political

  The government prefers to invest in low-cost, high-return projects. Because of these advantages, the project will not be opposed by the state or the government, as it will contribute to technology development to a certain extent.

- Legal

  As information in digital form is more susceptible to plagiarism and copying, the issues involved in this project are more likely to be related to copyright. The project will ensure that the resources used are legal (from regular and permitted sources). The project will not be public, the version control platform (GitHub) will be private, and any future commercial activity will be legal and compliant. In addition, as it relates to drones, it is necessary to comply with the laws and regulations about drone [34], which will need to be reflected in this project (e.g. no-fly zones, altitude restrictions).

6 Risk Register

A higher value means more **likely to happen** or has a more severe **impact**. A higher **Risk Score** means more attention is needed on that risk.

Risk	Mitigation	Likelihood	Impact	Risk score
Software (game engine) fail/corrupts, loose unsaved progress	Save the project frequently. A version control platform should be used (GitHub, Plastic SCM) to continually update.	4	3	12
losing project files	Frequent Backup is mandatory	1	4	4
Late Support from Supervisor	Take the initiative to follow up and communicate with the supervisor regularly.	2	3	6
Control devices' late support	At least one of the control hardware should be available (Gamepad, keyboard & mouse); use the existing devices first.	2	4	8
The computer broke/lost	Backup project files to cloud/disk every day (GitHub, OneDrive/hard disk)	1	4	4
Combination of ROS &Unity failed	More research and self-study on ROS and Unity. ask the supervisor for help(permitted)	1	4	4

Table 6.1: Risk register table, Risk score = Likelihood (1∼4) x Impact (1∼4)

7 Project Time Line

Time management will be divided into three parts, detailed information as shown in Figure 7.1

1. The first part is the project allocation, which is passed.

2. The second part is the development of the project. Software development and integration with hardware will be implemented and perfected in this section; some tasks are overlapped, which means these tasks should be implemented synchronously; these tasks are not interdependent but can be done simultaneously to improve efficiency and ensure that the project stays on the track since each element of the project has the similar and correct progress.

3. The final part is writing the dissertation, which could be started once the fundamental and core project goals have been implemented.

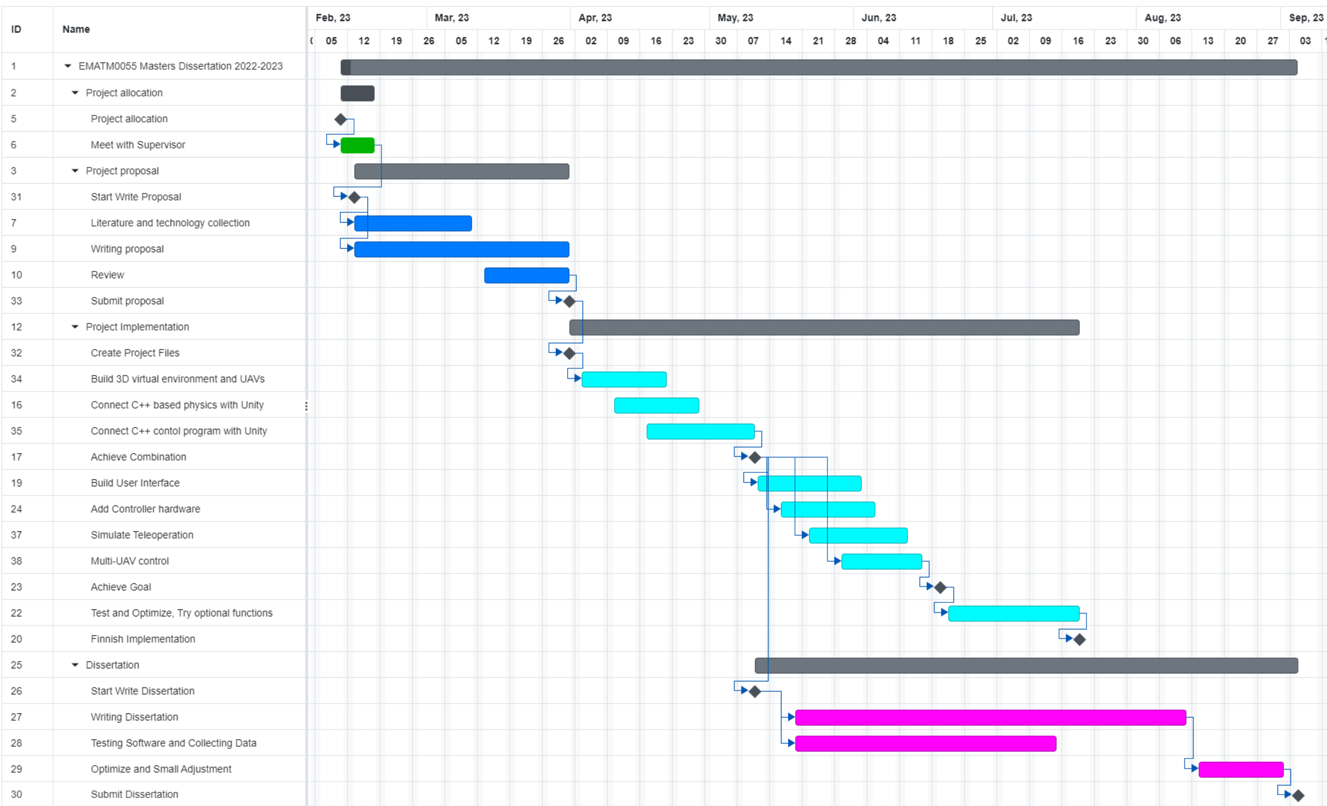


Figure 7.1: Project Time-Line Gantt-chart

References

[1] L. R. Salinas, J. Gimenez, D. C. Gandolfo, C. D. Rosales, and R. Carelli, Unified motion control for multilift unmanned rotorcraft systems in forward flight, *IEEE Transactions on Control Systems Technology* [online] 2023, pp. 1–15, 2023. DOI: 10.1109/TCST.2023.3240541.

[2] U. Technologies, *Unity user manual 2021.3 (lts)*. available from: https://docs.unity3d.com/Manual/index.html.

[3] M. Pinol, *Simulating robots with ros and unity*, Oct. 2022. available from: https://resources.unity.com/unitenow/onlinesessions/simulating-robots-with-ros-and-unity.

[4] J. Gimenez, D. C. Gandolfo, L. R. Salinas, C. Rosales, and R. Carelli, Multi-objective control for cooperative payload transport with rotorcraft uavs, *ISA transactions*, vol. 80 2018, pp. 491–502, 2018.

[5] L. R. Salinas, E. Slawiñski, and V. A. Mut, Complete bilateral teleoperation system for a rotorcraft uav with time-varying delay, *Mathematical Problems in Engineering*, vol. 2015 2015, 2015.

[6] J. Collins, S. Chand, A. Vanderkop, and D. Howard, A review of physics simulators for robotic applications, *IEEE Access*, vol. 9 2021, pp. 51 416–51 431, 2021.

[7] J. Linnell, *Robots are hard, game engines are not: Why we built our own simulator using unity*, Oct. 2022. available from: https://resources.unity.com/ai-ml-content/robots-are-hard-game-engines-are-not-why-we-built-our-own-simulator-using-unity.

[8] Y. Tan and Z.-y. Zheng, Research advance in swarm robotics, *Defence Technology*, vol. 9, no. 1 2013, pp. 18–39, 2013.

[9] J. Irizarry, M. Gheisari, and B. N. Walker, Usability assessment of drone technology as safety inspection tools, *Journal of Information Technology in Construction (ITcon)*, vol. 17, no. 12 2012, pp. 194–212, 2012.

[10] M. Hassanalian and A. Abdelkefi, Classifications, applications, and design challenges of drones: A review, *Progress in Aerospace Sciences*, vol. 91 2017, pp. 99–131, 2017.

[11] K. Mohammadi, M. Jafarinasab, S. Sirouspour, and E. Dyer, "Decentralized motion control in a cabled-based multi-drone load transport system," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2018, pp. 4198–4203. DOI: 10.1109/IROS.2018.8593952.

[12] D. K. Villa, A. S. Brandao, and M. Sarcinelli-Filho, A survey on load transportation using multirotor uavs, *Journal of Intelligent & Robotic Systems*, vol. 98 2020, pp. 267–296, 2020.

[13] S. Barawkar and M. Kumar, Force-torque (ft) based multi-drone cooperative transport using fuzzy logic and low-cost and imprecise ft sensor, *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering* 2023, p. 09 544 100 231 153 686, 2023.

[14] T. R. R. Staff, *Small drones team up to carry heavy packages*, Mar. 2021. available from: https://www.therobotreport.com/drones-team-up-to-carry-heavy-packages/.

[15] *Teleoperation of unmanned rotorcrafts transporting a cable-suspended load - Unity with Oculus Rift*. YouTube, Feb. 2019. available from: https://www.youtube.com/watch?v=keUsXU_zHGw.

[16] S. Shah, D. Dey, C. Lovett, and A. Kapoor, "Airsim: High-fidelity visual and physical simulation for autonomous vehicles," in *Field and Service Robotics: Results of the 11th International Conference*, Springer, 2018, pp. 621–635.

[17] Y. Song, S. Naji, E. Kaufmann, A. Loquercio, and D. Scaramuzza, "Flightmare: A flexible quadrotor simulator," in *Proceedings of the 2020 Conference on Robot Learning*, 2021, pp. 1147–1157.

[18] R. Andrade, G. V. Raffo, and J. E. Normey-Rico, "Model predictive control of a tilt-rotor uav for load transportation," in *2016 European Control Conference (ECC)*, IEEE, 2016, pp. 2165–2170.

[19] K. N. Tahar and A. Ahmad, An evaluation on fixed wing and multi-rotor uav images using photogrammetric image processing, *International journal of computer and information engineering*, vol. 7, no. 1 2013, pp. 48–52, 2013.

[20] A. Y. Mersha, S. Stramigioli, and R. Carloni, On bilateral teleoperation of aerial robots, *IEEE Transactions on Robotics* [online], vol. 30, no. 1 2014, pp. 258–274, 2014. DOI: `10.1109/TRO.2013.2281563`.

[21] A. Franchi, C. Secchi, H. I. Son, H. H. Bulthoff, and P. R. Giordano, Bilateral teleoperation of groups of mobile robots with time-varying topology, *IEEE Transactions on Robotics* [online], vol. 28, no. 5 2012, pp. 1019–1033, 2012. DOI: `10.1109/TRO.2012.2196304`.

[22] M. Moniruzzaman, A. Rassau, D. Chai, and S. M. S. Islam, Teleoperation methods and enhancement techniques for mobile robots: A comprehensive survey, *Robotics and Autonomous Systems*, vol. 150 2022, p. 103 973, 2022.

[23] C. Bartneck, M. Soucy, K. Fleuret, and E. B. Sandoval, "The robot engine — making the unity 3d game engine work for hri," in *2015 24th IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN)*, 2015, pp. 431–437. DOI: `10.1109/ROMAN.2015.7333561`.

[24] U. Technologies, *Robotics simulation*. available from: `https://unity.com/solutions/automotive-transportation-manufacturing/robotics`.

[25] A. Juliani, V.-P. Berges, E. Vckay, *et al.*, Unity: A general platform for intelligent agents Sep. 2018, Sep. 2018.

[26] H. Choi, C. Crump, C. Duriez, *et al.*, On the use of simulation in robotics: Opportunities, challenges, and suggestions for moving forward, *Proceedings of the National Academy of Sciences*, vol. 118, no. 1 2021, e1907856118, 2021.

[27] J. Gregory, *Game engine architecture*. crc Press, 2018.

[28] A. Salomão, F. Andaló, and M. L. H. Vieira, "How popular game engine is helping improving academic research: The designlab case," in *Advances in Human Factors in Wearable Technologies and Game Design: Proceedings of the AHFE 2018 International Conferences on Human Factors and Wearable Technologies, and Human Factors in Game Design and Virtual Environments, Held on July 21–25, 2018, in Loews Sapphire Falls Resort at Universal Studios, Orlando, Florida, USA 9*, Springer, 2019, pp. 416–424.

[29] M. Quigley, K. Conley, B. Gerkey, *et al.*, "Ros: An open-source robot operating system," in *ICRA workshop on open source software*, Kobe, Japan, vol. 3, 2009, p. 5.

[30] B. V. Kasulaitis, C. W. Babbitt, R. Kahhat, E. Williams, and E. G. Ryen, Evolving materials, attributes, and functionality in consumer electronics: Case study of laptop computers, *Resources, conservation and recycling*, vol. 100 2015, pp. 1–10, 2015.

[31] *Delivery drones and the environment*, Mar. 2021. available from: https://www.eea.europa.eu/publications/delivery-drones-and-the-environment.

[32] J. Koiwanit, Analysis of environmental impacts of drone delivery on an online shopping system, *Advances in Climate Change Research* [online], vol. 9, no. 3 2018, pp. 201–207, 2018, ISSN: 1674-9278. DOI: https://doi.org/10.1016/j.accre.2018.09.001. available from: https://www.sciencedirect.com/science/article/pii/S1674927818300261.

[33] R. Luppicini and A. So, A technoethical review of commercial drone use in the context of governance, ethics, and privacy, *Technology in Society* [online], vol. 46 2016, pp. 109–119, 2016, ISSN: 0160-791X. DOI: https://doi.org/10.1016/j.techsoc.2016.03.003. available from: https://www.sciencedirect.com/science/article/pii/S0160791X16300033.

[34] *Flying drones and model aircraft*. available from: https://register-drones.caa.co.uk/.