UNIVERSITY OF WARWICK

UAVSI

UNMANNED AERIAL VEHICLE SWARM INTELLIGENCE

Progress Report

Antonio Brito

1 Background

The uses for autonomous systems are ever-growing, with applications in logistics, agriculture, remote sensing and situational awareness[1][2]. The aim of this project is to explore the use of autonomous systems in a defensive manner, specifically in the context of a swarm of Unmanned Aerial Vehicles (UAVs) in a hostile environment. The project will explore the use of swarm intelligence to ensure the survivability of the swarm in a hostile environment, where the swarm is tasked with reaching a goal location whilst avoiding obstacles and hostile entities.

In the final report, we will also look back on findings to explore the parallels between the agents' behaviour and expected human behaviour when acting in swarms to evaluate the feasibility of future research on intelligent systems. The overall goal of the project is to combine existing technology in a novel way; in a field of research lacking open source resources.

The aim of this report is to document the progress made so far in the project, as well as outlining changes to the timetable for the project.

2 Progress

The majority of the work completed so far has been research and development of the physics model and the simulation of initial swarm behaviour, based upon Reynolds' *Boids*[3].

The completed work can be split into several subsections as follows:

2.1 Quadcopter Operation

A model simulating the movement and physics of a quadcopter operating in an Earth-like environment has been developed. This has been achieved by simulating a quadcopter as a rigid body, with four *'propellers'*[1] generating thrust. At this stage, some constants have been assigned to the model in order to simulate operation in an Earth-like environment. These constants may be refined as the project progresses if needed. The table of these constants is seen in the table below.

To gain familiarisation with the *Unity Engine*, the initial aim was to operate the quadcopter using keyboard controls, rather than autonomously. The design of the control system and model was largely based on *Luukkonen's* research in the field[4] alongside some high-level explanatory reading in *IJREAM*[2][5].

The keyboard operation was based on 8 axes of control. These are summarised in Table 1.

Operation	**Key**
Positive Pitch	W
Negative Pitch	S
Roll Left	A
Roll Right	D
Yaw Clockwise	L
Yaw Anticlockwise	J
Ascend	I
Descend	K

Table 1: Keyboard Drone Operation

This then allowed for a control system to be tested. The aim of this system was to allow for the drone to be controlled in a stable manner, with the drone returning to a stable position when no controls were being applied. This was achieved by using a *PID* controller[6, p72-73]. Whilst there

[1] For the purposes of the simulation, the propellers do not rotate. Thrust is simulated as an upward force.

[2] International Journal for Research in Engineering Application & Management

was some success from the literature, the implementation of the controller largely came from online tutorials, namely from *Carbon Aeronautics*[7].

The *PID* controller was then tuned using an empirical method. This was done by setting the P constant to a high value, and then reducing it until the drone was stable. The I and D constants were then tuned in a similar manner. The provisional final values of the constants are shown in Table 2.

Control Operation	K_p	K_i	K_d
Thrust	6	5	2
Pitch	10	10	2
Roll	10	10	2
Yaw	10	10	2

Table 2: Tuning Constants

2.2 Swarm

The initial work surrounding the operation of the swarm has been completed. This consists of the three separate *Boids* behvaiours - alignment, cohesion and separation[3]. Combining these three behvaiours and tuning their respective influences for effective swarm operation has not yet been completed. This has been reflected in the outline of the revised project timetable.

Additionally, the mathematical model of the swarm's operation itself has not been completed. Currently, high-level methods are being used to control the individual behaviours, namely *Unity's* `transform.worldToLocalMatrix` method. This is used to create a transformation matrix in the local coordinate frame of the agent, which is then used to calculate the desired direction of movement. This is not ideal, as it is not a true mathematical model of the swarm's operation. Ideally, this will be expanded to utilise quarternions to model the rotation of an agent. So far, *3Blue1Brown*'s interactive guide has proven useful in understanding the mathematics behind quarternions[8]. The complexity in understanding quarternions has been reflected in the revised timetable.

2.3 Environment

The initial project noted the development of the environment would be partly done in parallel with the physics model and completed before the work on pathfinding. With some discussions, it was decided that as the environment may partly constitute an obstacle, the major work on creating a complex and noisy environment would be completed after the work on pathfinding, at the stage of working on collision avoidance. This is reflected in the revised timetable.

In place of this, an initial environment was completed, formed of a simple ground plane. There are currently plans to create a randomly assigned goal location within the dimensions of the plane to allow for the development and testing of the pathfinding algorithms.

3 Project Management

Generally, the management of the project so far has been successful. As with any project, there have been delays, which have been mitigated by effective risk planning and management. The delays are explained alongside the revised timetable in Section 4. In spite of this, the project's development is consistent, largely in part due to being held to account by regular, usually weekly, meetings with the project supervisor.

4 Future Plan

At this stage, it is useful to outline the timetable changes. These are seen in Figure 1.

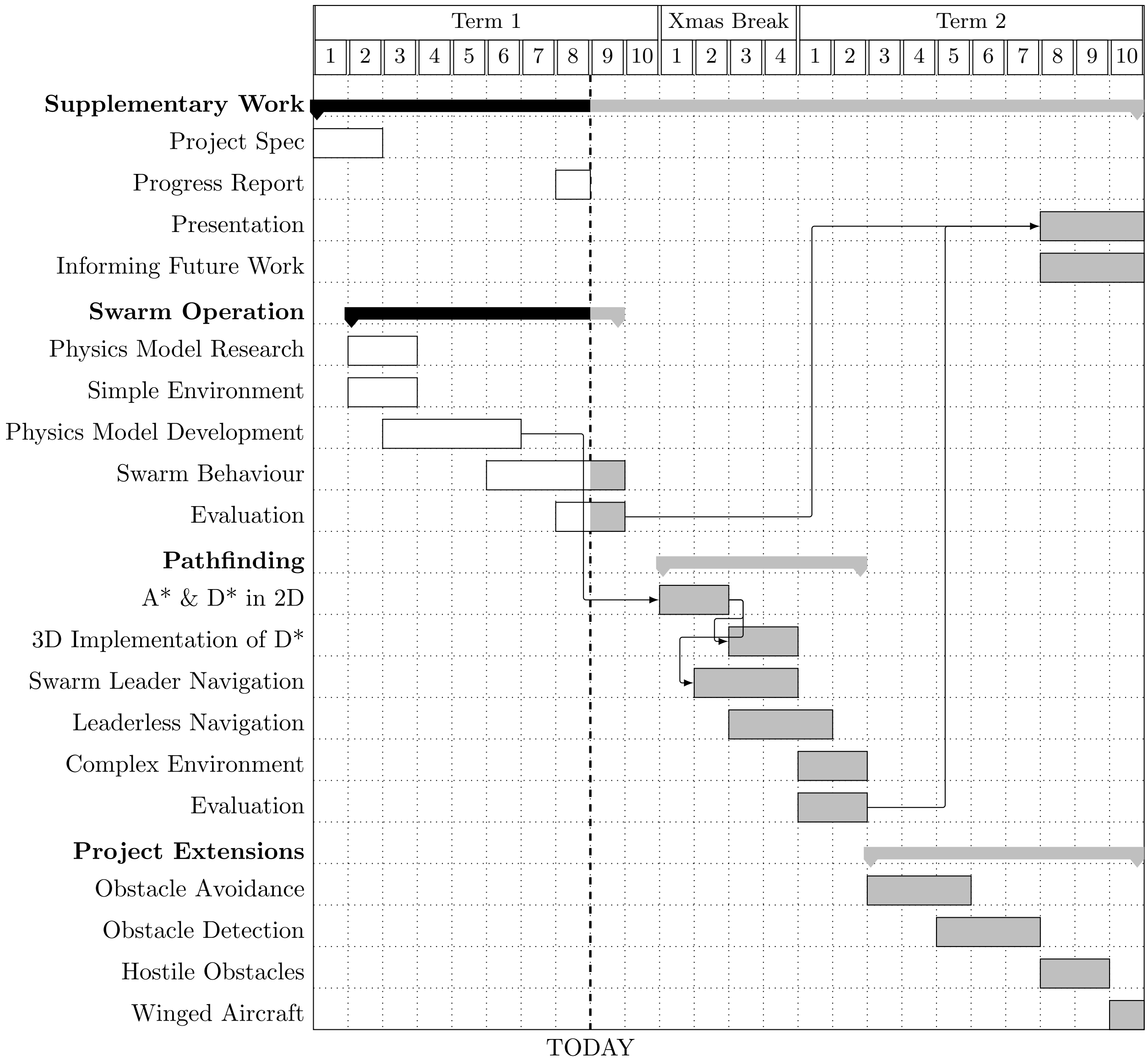

Figure 1: Revised Gantt Chart. Numbers shown are week numbers.

The changes to the timetable can be grouped and summarised as the following:

4.1 Swarm & Agent Physics

As explained in Section 2, the development of the swarm model has proven more complex than expected. As such, the expected timeframe for completion of this group of tasks (Swarm Operation) has been extended by 3 weeks. This is to allow for the completion of the swarm model, as well as the evaluation of the swarm's operation, which has mostly been completed in parallel, partly owing to the delays in the swarm model's development.

Secondly, a new task has been added to the chart - *Swarm Behaviour*. This is a subset of the *Physics Model* task, with a greater focus on getting the swarm to act in a stable manner.

4.2 Change to Environment Development

The decision to split the stage of development focusing on the environment model into two distict stages, namely *Simple Environment* and *Complex Environment* in Fig. 1 has warranted the need for a new task in the chart to achieve the latter half of the split. This has meant a slight shift in the *Pathfinding* task group.

4.3 Project Extensions

Lastly, the timeframes for the tasks within the project extension task group have been shortened. This is to allow for more time spent on earlier, more vital tasks. This was accounted for in the risk assessment, as part of the project specification.

5 Ethics

As noted previously in the project specification, there are no ethical concerns with the project. The project is purely a simulation, and as such, there are no concerns with human involvement or the usage of personal data. However, at this stage, it should be noted that the project deals with artificial intelligence, namely autonomous systems. Notably, the Students' Union passed a motion in 2023 which aimed to lobby the University to *'form an internal Research Ethics Committee for Computer Science and AI, in order to assess and mitigate the risk that university dual-use research could be applied for unintended malicious uses or incorporated in harmful weapons systems, such as autonomous weapons systems'*[9]. This is a reccomendation that should be considered by the department.

6 Conclusion

Generally, we note that the project is on track. The main changes in the timetable reflect minor delays and changes in the order of tasks. As such, the project is still expected to finish on time.

References

[1] G. Muchiri and S. Kimathi, "A review of applications and potential applications of uav," in *Proceedings of the Sustainable Research and Innovation Conference*, 2022, pp. 280–283.

[2] J. L. Drury, L. Riek, and N. Rackliffe, "A decomposition of uav-related situation awareness," in *Proceedings of the 1st ACM SIGCHI/SIGART Conference on Human-Robot Interaction*, ser. HRI '06. New York, NY, USA: Association for Computing Machinery, 2006, p. 88–94. [Online]. Available: https://doi.org/10.1145/1121241.1121258

[3] C. W. Reynolds, "Flocks, herds and schools: A distributed behavioral model," *SIGGRAPH Comput. Graph.*, vol. 21, no. 4, p. 25–34, aug 1987. [Online]. Available: https://doi.org/10.1145/37402.37406

[4] T. Luukkonen, "Modelling and control of quadcopter," *Independent research project in applied mathematics, Espoo*, vol. 22, no. 22, 2011.

[5] O. Tatale, N. Anekar, S. Phatak, and S. Sarkale, "Quadcopter: design, construction and testing," *International Journal for Research in Engineering Application & Management*, vol. 4, pp. 1–7, 2018.

[6] G. Szafranski and R. Czyba, "Different approaches of pid control uav type quadrotor," in *International Micro Air Vehicle conference and competitions 2011 (IMAV 2011),'t Harde, The Netherlands, September 12-15, 2011.* Delft University of Technology and Thales, 2011.

[7] C. Aeronautics, "Mathematically design a drone pid controller," 2023. [Online]. Available: https://www.youtube.com/playlist?list=PLeuMA6tJBPKvqIveRYTqwBjjXLblVpeoh

[8] 3Blue1Brown, "Quaternions and 3d rotation, explained interactively," Oct 2018. [Online]. Available: https://www.youtube.com/watch?v=zjMuIxRvygQ&ab_channel=3Blue1Brown

[9] "Asv motion to take a stance against the development of autonomous weapons," 2023. [Online]. Available: https://www.warwicksu.com/referenda/motion/131/374/

Appendix: Project Specification Overleaf

UNIVERSITY OF WARWICK

UAVSI

UNMANNED AERIAL VEHICLE SWARM INTELLIGENCE

Project Specification

Antonio Brito

1 Abstract

This project explores simulating methods of unmanned aerial vehicles (UAVs) moving, pathfinding and completing tasks in bird-like groups (swarms), initially based off of Boids theory [1].

2 Background

This project combines previous research from several disciplines into one. Looking across these fields, we have **Swarm Intelligence**, which explores a group of agents interacting locally with one another and with their environment[2]. Boids theory is some of the initial work that fits into this field. This project aims at augmenting this with modern swarm intelligence metaheuristics. Aside from this, the aim is to explore the data analysis process from the point of sensing, tracking, state estimation, through to processing and decision making, autonomously. Lastly, we look at managing a UAV system in three dimensions using classical flight mechanics.

In the final report, we will also look back on findings to explore the parallels between the agents' behaviour and expected human behaviour when acting in swarms to evaluate the feasibility of future research on intelligent systems. The overall goal of the project is to combine existing technology in a novel way; in a field of research lacking open source resources.

3 Objectives

The aim of the project can be split down into four distinct sections, grouped by similarity and priority. For all objectives, several implementations will be explored and compared.

3.1 Swarm Operation

The aim here is to simulate a group of identical nodes (UAVs) operating cohesively as a swarm. This objective is further broken down into simulating the physics of the model and the environment as such:

3.1.1 Physics Model

As we are concerned with a 3D environment, the model of the agents will fit such an environment. The agents will be modelled as quadcopters[3], rather than winged aircraft, for simplicity. This is because of the following considerations[1]:

- High maneuverability is key in swarms to ensure high rates of survivability.
- Aircraft position can be fixed, allowing for greater analysis time.
- Vertical take-off and landing (VTOL) is possible.

As a trade-off, the quadcopter model will be slower and will likely be less stable in real-life scenarios. However, we will explore extending the project to looking at conventional winged aircraft flight. Using the Boids model, we will implement the three key rules for emergent swarm behaviour:

- **Node Separation**
  Simulating the ability for nodes to keep some minimum distance from other nodes, to prevent collisions.
- **Node Alignment**
  Simulating nodes tending to have similar velocity vectors to nearby nodes, such that they travel in the same direction.

[1] Quadcopter vs Drone: What's the Difference?

- **Node Cohesion**
  Simulating nodes moving towards the centre of nearby nodes to uphold the 'pack'-like behaviour.

Ideally, after the physics model is created, a working simulation of a group of agents will be demonstrated. The number of agents will be dynamic, as this may later affect optimisations. At this stage of development, the aim will be to have minimal to no collisions between agents with a swarm size of **8-16** drones. The relationship between swarm size and number of collisions will be modelled.

3.1.2 Environment Model

The environment will be continous and three dimensional. The aim will be to have a finite world size, with randomly generated terrain according to a continous heightmap, likely using Perlin noise[4]. Agents will have a start location and a goal location, generated randomly at every iteration of the simulation.

3.2 Pathfinding

To introduce and compare several global path planning algorithms, we will first use a 2D map to model both the **A***[5] and **D* Lite**[6] algorithms for individual node pathfinding. We will then explore a 3D implementation of the more useful algorithm for the project, likely D*, alongside a case study for the algorithm - the Mars Exploration Rovers[7]. Noticeably, a large part of the development timetable in Section 5 is devoted to this 3D implementation. This is due to a lack of resources avaliable for this specific implementation.

We will look at three separate implementations of pathfinding:

- **The Naive, Leaderless Approach**: Does chaos occur without an elected leader, leaving each agent to think for themselves?
- **A Combined Approach for Swarm Operation & Pathfinding**: We will attempt to model swarm pathfinding with collision avoidance and cohesion heuristics, such that the edge weights that bring nodes too close or too far to other nodes are high-cost.
- **D* with Leadership Behaviours**: Does electing a leader reduce computational power requirements at a trade-off for survivability?

For all of these implementations, we will consider the computational power required for simulation, aiming to minimise this so that larger numbers of agents can exist.

3.3 Obstacle Avoidance

Assuming a suitable method of pathfinding is found, the next objective will be to simulate an environment with obstacles and model obstacle avoidance within the swarm. The obstcle-environment relationship is classified by the following characteristsics:

- Obstacles are randomly generated from a set of types (such as Floating Cube, Mountain, Building).
- The size of obstacles will be random, within a Gaussian distribution.

This objective can be broken down into two parts:

3.3.1 Perfect Representation of the Environment

Initially, the goal is to ensure a high survivability rate amongst the swarm when given a perfect representation of the location and classification of obstacles. The project will explore the use of two main avoidance mechanisms:

1. Naive D* Avoidance using edge-weights
2. Simulated Magnetic Fields for Collision Avoidance[8]

3.3.2 Incomplete Representation of the Environment

The focus will then shift onto how the swarm handles receiving incomplete data regarding obstacles, such as position without a type classification, or vice versa. The goal is to explore autonomous behaviour when considering aspects of probability. At this stage, we consider obstacle density as another factor; in less challenging terrain, avoidance is easier.

3.4 Obstacle Detection

An extension to the project will consist of exploring methods of obstacle detection. This objective, as well as the next objective, begins to look at a simulated implementation of state estimation & tracking models. The inital steps for this objective will be looking at methods of sensing, primarily LiDAR[9], to detect obstacle positions.

Conequently, we will explore how sensor placement within the swarm affects survivability and whether an increased number of sensors leads to better results (using data fusion techniques).

3.5 Hostile & Moving Obstacles

A final possible extension will be looking at the detection and avoidance of hostile and moving obstacles. To complete this, hostile obstacles are obstacles with the capability of firing projectiles or signal jamming across some unknown (random) radius. Moving obstacles traverse the terrain linearly in a given (random) direction. This could be further extended to implement random non-linear walks.

The aim of this objective is to explore autonomous behaviour in environments where predictions need to occur when considering danger travelling towards the swarm.

4 Methods

The proejct will be completed in Unity due to the lack of licensing requirements, ease of use and scripting support[10], likely using C++. Consequently, the simulation will be a Unity3D environment. The codebase in its entirety, including source code and version control, will be managed by a cloud-based GitHub repository[11].

The **waterfall** software methodology will be used in development. This is evident in the timetable in Section 5. The waterfall approach lends itself to structured stages of development, where one stage leads naturally into the next. For the purposes of this project, these stages are outlined in bold in the timetable in Section 5.

For this approach, the model splits the development process into the following sections, specific to this project:

1. **System** - the project specification outlining the tasks to be completed with rough timelines.
2. **Analysis** - a section in the final report detailing the models used and techniques compared.
3. **Coding** - the development of the system.
4. **Testing** - validating the system to ensure the project's objectives have been met.

5 Timetable

Generally, the aim is to split the objectives above to be completed according to the timetable below:

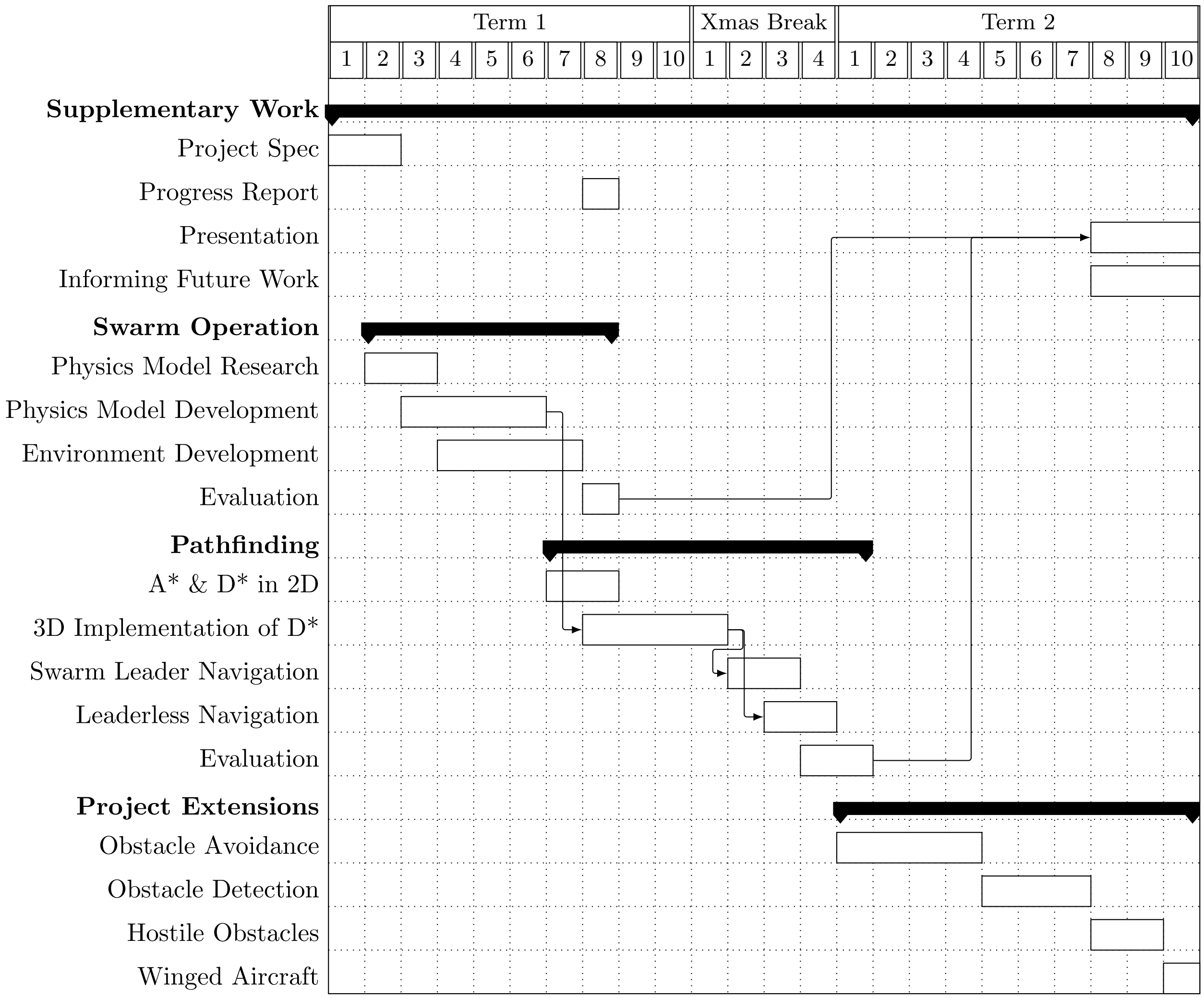

Figure 1: Gantt Chart showing main objectives. Numbers shown are week numbers.

Noticeably, the timetable leaves a large chunk of time between the scheduled completion of essential objectives (Term 2 Week 1) and the presentation (Term 2 Week 9). This is to leave a margin of error induced by risks and unexpected events, as discussed in the next section.

6 Risks & Issues

The major risks to the project are as follows:

1. **Stages in the project fall behind schedule.**
   This may occur when a critical task takes too long to complete. The solution to this issue is to ensure alternatives exist at all critical points and the timetable lends itself to having leeway.

2. **Insufficient computational power.**
Naturally, simulation lends itself to requiring high computing specifications. A side aim of the project is optimising the simulation environment to ensure it is able to be ran on my personal computer. If all else fails, a wise option is to consider the Department's access to computational resources.

3. **Project work is lost and/or unretrievable.**
As all development will occur on a personal computer, it is essential to ensure backup copies exist elsewhere. Hence, all written reports, specifications and source code will be backed up to GitHub as a minimum.

4. **Insufficient findings for a complete report.**
There is often more to write about regarding failures than successes, as all other reasonable options must be exhausted before declaring failure! No useful product is guaranteed, but findings certainly are.

5. **Libraries or resources become unavaliable during the project process.**
Whilst this is low risk, to mitigate this, local copies will be used wherever possible and backed up alongside the codebase.

6.1 Ethical Issues

A research project involving autonomous technology comes with vital ethical considerations. All research will be conducted in line with the Department's ethical guidelines[2], as no human imvolvement is required.

It is important to ensure there is no ethical ambiguity for the purposes of the project. As such, the intended research goals as detailed in the problem statement will be kept to those without ethical ambiguity or favouring. The aim through this project is to inform future research for using intelligent autonomous systems defensively.

[2] Warwick DCS Ethical Guidelines

References

[1] C. W. Reynolds, "Flocks, herds and schools: A distributed behavioral model," *SIGGRAPH Comput. Graph.*, vol. 21, no. 4, p. 25–34, aug 1987. [Online]. Available: https://doi.org/10.1145/37402.37406

[2] J. Hu, A. E. Turgut, T. Krajník, B. Lennox, and F. Arvin, "Occlusion-based coordination protocol design for autonomous robotic shepherding tasks," *IEEE Transactions on Cognitive and Developmental Systems*, vol. 14, no. 1, pp. 126–135, 2022.

[3] M. S. Khan, "Quadcopter flight dynamics," *International Journal of Scientific & Technology Research*, vol. 3, pp. 130–135, 2014. [Online]. Available: https://api.semanticscholar.org/CorpusID:16365513

[4] K. Perlin, "An image synthesizer," *SIGGRAPH Comput. Graph.*, vol. 19, no. 3, p. 287–296, jul 1985. [Online]. Available: https://doi.org/10.1145/325165.325247

[5] P. E. Hart, N. J. Nilsson, and B. Raphael, "A formal basis for the heuristic determination of minimum cost paths," *IEEE Transactions on Systems Science and Cybernetics*, vol. 4, no. 2, pp. 100–107, 1968.

[6] S. Koenig and M. Likhachev, "D*lite." 01 2002, pp. 476–483.

[7] J. Carsten, A. Rankin, D. Ferguson, and A. Stentz, "Global path planning on board the mars exploration rovers," in *2007 IEEE Aerospace Conference*, 2007, pp. 1–11.

[8] F. Wu, A. Vibhute, G. S. Soh, K. L. Wood, and S. Foong, "A compact magnetic field-based obstacle detection and avoidance system for miniature spherical robots," *Sensors*, vol. 17, no. 6, p. 1231, May 2017. [Online]. Available: http://dx.doi.org/10.3390/s17061231

[9] A. Devos, E. Ebeid, and P. Manoonpong, "Development of autonomous drones for adaptive obstacle avoidance in real world environments," in *2018 21st Euromicro Conference on Digital System Design (DSD)*, 2018, pp. 707–710.

[10] I. Buyuksalih, S. Bayburt, G. Buyuksalih, A. Baskaraca, H. Karim, and A. Rahman, "3d modelling and visualization based on the unity game engine - advantages and challenges," vol. IV-4/W4, 11 2017, pp. 161–166.

[11] J. Blischak, E. Davenport, and G. Wilson, "A quick introduction to version control with git and github," *PLoS computational biology*, vol. 12, p. e1004668, 01 2016.