University of Bristol

COMSM3201

MSc Project Computer Science

Risk-based Triggering of Bio-inspired Self-Preservation to Protect Robots from Threat

Candidate No.:
58555

August 28, 2018

Abstract

Abstract

Contents

1 Glossary

Delivery Robot: This refers to the autonomous ground-based delivery robot being considered the “prey” in this project. The delivery robot discussed in this paper is assumed to be based on the Morse ATRV.

Drone: The drone is the autonomous robot being considered the predator throughout this project. It is an air-borne drone based on the Morse Quadrotor.

Morse simulation software: The Modular OpenRobots Simulation Engine, is an open source generic simulator for academic robots.

Cautious drone: A drone whose priority is returning to its refuge before it runs out of battery.

Persistent drone: A drone who has infinite battery charge and who will continue until it is successful.

Simulated environment: This refers to the virtual environment in which the project will be carried out. Both robots and their surrounding will be simulated i.e. there will be no physical robots used or damaged in the course of this experiment.

Refuging time: The length of time the prey remains in a refuge in an attempt to avoid the predator.

Flight initiation distance: This is the nearest the prey will allow a perceived predator to approach before taking flight.

Alert distance: The distance at which a non-fleeing prey begins to monitor the actions of a perceived predator. This directly precedes flight initiation distance in attack situations.

Scanning rate: The frequency with which the prey assesses the predator’s behaviour to monitor for an impending attack.

Finding refuge: The prey finds somewhere to hide into which the predator cannot advance or in which they become invisible to the predator.

2 Introduction

This section will outline the inspiration for this project, its aims and objectives, the project deliverables, the added value the project has provided the Computer Science community and the scope of the project.

2.1 Project Background

Over the course of the summer of 2017 a paper titled Risk-based Triggering of Bio-inspired Self-Preservation to Protect Robots from Threat was published based on a project conducted by a student of Computer Science in the University of Bristol [1]. This project investigated the best strategy for a delivery robot to adopt when being pursued by two different types of mal-intended drones. The project carried out a series of tests using cautious and persistent drones, both of whom were aggressively pursuing the robot.

The delivery robot was either programmed with or without risk-based triggering in order to assess if a robot who responded to the threat of a drone would have a better chance of surviving the interaction than one who continued on its mission, regardless [1]. Moreover, delivery robots who were programmed with risk-based triggering were either programmed with some combination of self-preservation behaviours (e. g. fleeing and seeking refuge) or simply fleeing.

The results of the project, presented in the paper found that robots programmed with risk-based triggering had a better rate of success than those without [1]. It was also established that in the case of a ground-based delivery robot being pursued by a cautious drone, the ideal evasion strategy was a combination of fleeing and refuging. However, the ability of a robot to balance both these self-preservation behaviours and other long-term survival necessities was not investigated. The present project has contributed to the world of Computer Science by doing so.

2.2 Aims and Objectives

Over the course of this project, the aim was to survey the literature on the ways animals in the wild vary their behaviour when there is a risk of being attacked. The project assessed the methods by which animals alter their behaviour before and during the course of a predatory attack with a particular focus on flight initiation distance and refuge time. Based on these findings, delivery robots were programmed with such behaviours over a number of simulated tests, with the aim of establishing how the behaviours impacted the robot's success rate and efficiency. In order to achieve these aims, a set of tangible objectives for this project was drawn up and were as follows:

- Conduct a thorough review of literature sources, presenting the most successful survival traits in animals and existing survival technology for robots, in order to establish the most biologically sound methods of escape from pursuit.
- Create a proposal for feasible anti-vandalism or damage preventive safety mechanisms (software based) in delivery robot-drone relationships, that complement and preserve human safety.
- Simulate all proposed mechanisms in Morse allowing a full evaluation of them, to demonstrate the preservation of both human safety and the robot's safety in the highest degree possible.

2.3 Testing, Evaluation and Deliverables

This section will lay out what was being tested in this project, what the controls were, what the evaluation method was and the deliverables that were seen as a result of this project set up.

2.3.1 Testing

The behaviours being tested in this project were:

- Flight initiation distance
- Refuging time

These behaviours' affect on the following were observes:

- Mission time
- Mission success
- Robot battery level
- Robot ability to reach refuge

In order to ensure fairness, the experiments were be conducted under strict conditions. The following measures were employed as a set of controls:

Features of the delivery robot were be consistent throughout: The robot used was the built in ATRV model in Morse. This model was considered reasonably similar to the Starship Technologies delivery robot in terms of dimensions. The Starship Technologies robot weighs 50lbs unloaded and is capable of transporting 22lbs. There was not time to develop the standard model to further resemble the Starship Technologies robot, although this could be useful in a continuation of this study with fewer time constraints.

Features of the drone were be the consistent throughout: Again, it was necessary to use the built in Quadrotor drone, provided in Morse simulation. This was expected to have maximum load bearing capabilities of up to 99lbs (based on the Harris Aerial HX8 Power XXL). The drone, in each test, were be cautious, thus had a finite amount of battery and was programmed to only pursue for as long as it has sufficient battery to subsequently return to base.

Testing standards: Two versions of the same robot were initially tested on, each programmed with different sets of behaviours. These were extreme cases with one displaying extremely cautious behaviours and one showing extremely reckless behaviours. Each robot was tested roughly one hundred times and the result of each noted.

2.3.2 Evaluation

This project sought to uncover the impact that flight initiation distance and refuging time had on four different criteria, each of which was captured at the end of every test. These four criteria were:

- Whether or not the robot was captured
- Whether or not the robot attempted to hide
- The amount of time it took for the robot to either be captured or to reach its destination
- The robot's battery level when it either was captured or reached its destination

These four pieces of information served to illustrate the impact that flight initiation distance and refuging time have on the robot's success rate and efficiency.

2.3.3 Deliverables

Based on the testing protocol laid out above, the following deliverables were achieved in their listed order:

- Establishment of a thoroughly tested, working environment with both delivery robot and drone working to specification.
- Establishment of the impact that refuge time has on a robot's success rate and efficiency.
- Establishment of the impact that flight initiation distance has on a robot's success rate and efficiency.
- Graphical illustration of results that will show the relative success of each of the behaviours being tested.

2.4 Added Value

This section will outline how the project contributed to the field of bio-robotics.

Ongoing projects: This project explored the affects of a set of varying behaviours for self-preservation on the efficiency and success rates of delivery robots. By beginning an investigation into finding the ideal combination of flight initiation distance and refuging time, this project immediately added value to the ongoing investigation into bio-inspired risk avoidance in delivery robots. This particular strain of investigation is being conducted in the University of Bristol and this project constituted an integral part of that investigation.

Autonomous systems in delivery robots: This project contributed to work on autonomous systems and their use in delivery robots. By identifying behaviours that can be used by a delivery robot to avoid capture or damage, it will become possible for robots to ensure their own safety, thus avoiding the need for human intervention. Autonomous robots must be able to adapt to their environment and situation, as animals do, in order to be more useful than manual robots and risk aversion plays a large part in this.

Robot safety: This project contributed to general investigations into robot safety, a field that is slowly emerging. As autonomous robots are introduced into the world on a wider scale, both robot and human safety will be of utmost importance. While human safety has already been thoroughly investigated, robot safety is still an emerging field and research into the ideal, bio-inspired behaviours for a robot being pursued by a drone is yet to be completed.

Bio-robotics: This project contributed to the general field of bio-robotics which is becoming increasingly popular. As robots evolve, more and more often people are looking to biology for inspiration. This project defined which animal behaviours are the best for use in a ground-based delivery robot. Thus, it contributed to discussions on the most useful animals to draw inspiration from, when designing an autonomous delivery robot.

2.5 Hypothesis

Primary hypothesis: Less cautious self-defence mechanisms in robots, which is seen in lower flight initiation distance and refuge time, allow for increased efficiency and energy conservation.

Secondary hypothesis: More energy focused robots conserve more energy but get captured more frequently.

3 Literature Survey

This section outlines the investigation carried out into the literature surrounding bio-inspired self-preservation methodology to protect robots from threats. In summary:

- The first section will examine the general question of robot safety, with most papers focusing on safety for the benefit to humans. This section will be particularly helpful in providing a background in appropriate testing techniques.
- The second section will examine pre-existing studies on bio-inspired self-preservation and past attempts to simulate predator-prey relationships or similar scenarios using robots.
- Thirdly, a study of pre-existing work on animal behaviours in the wild will be conducted. This section will provide the bio-inspiration for this project and deserves extensive discussion. This literature survey will be concluded with an evaluation of the overall picture being presented by the papers.

3.1 Safety in robots

Over the course of robotic development, strategies have been put in place in order to ensure that robots operate in a safe manner. Human safety in the presence of robots has, for obvious reasons, been of vital importance in the growth and development of the field of robotics. With the emergence of autonomous robots, it is becoming equally necessary to develop safety mechanisms that ensure robot safety. This project will address the ability of autonomous delivery-robots to protect themselves when equipped with self-preservation techniques. This section outlines the development of robot safety and the need for increased work into self-preservation techniques.

3.1.1 Human safety

Human safety has been at the forefront of robotics development from the beginning [2]. As the authors of Environmental hazard analysis - a variant of preliminary hazard analysis for autonomous mobile robots argue, in situations where robots must interact with humans, extensive precautions must be met [3]. With human controlled robots, the task of ensuring that a robot is safe for use, can fall to the controller and they can be expected to ensure that the robot responds appropriately to all environments, tasks or obstacles encountered. However, with the emergence of autonomous robots, it is necessary to come up with a new solution to ensure that an autonomous robot will operate appropriately once released into an environment. The benefit of autonomous robots is that they can self-regulate and do not need human intervention to assess each situation. This has resulted in several attempts to come up with new criteria to ensure human safety around autonomous robots.

Dogramadzi et al have examined this issue and have looked at the benefits of using Environmental Survey Hazard Analysis (ESHA), which involves assessing non-mission interactions and threats that might ve associated with them12. However, this method was only successful when implemented on a small number of robots in a known environment [3]. The investigations into ESHA are somewhat simplistic in their approach, as is noted by the authors in their conclusions. The method requires the designer to prepare the robot for only one type of environment. Moreover, as also noted by the authors, there is no possibility that a designer would be capable of foreseeing every mission an autonomous robot might be required to perform over the course of their existence. Similarly, in the proposal put forward by Woodman, Winfield, Harper and Fraser of Bristol Robotics Laboratory in which it is suggested that a data set of potential hazards be gathered and used in a system that both verifies safety constraints and stops the robot from performing potential hazardous tasks, the robot designer must foresee every hazard that might be presented to the robot [4]. While the method is considered generally successful, its shortcoming is its reliance on human prediction of all appropriate risks and mitigation against them.

One method of ensuring human safety that has been used extensively in industrial settings is the shutting down of the robot when a human comes within a specific proximity. As is discussed in [5] this is only useful in scenarios where human-robot cooperation is not required. The issue of human-robot cooperation - i.e. the robot and human being able to work together, rather than working one at a time - is founded in the fact that robots do not yet understand human injuries and as such find it difficult to assess when they might be causing a human harm. This is an obstacle that must be overcome in order for autonomous robots to be capable of existing side-by side with humans. Indeed, imbuing robots with their own sense of self-preservation may well aid in allowing them to understand and assess the damage they can cause to living beings.

One of the key factors affecting human safety in human-robot interactions is robot socialisation. As autonomous robots become increasingly capable of understanding humans' social cues, their ability to understand non-verbal communication will improve drastically [6]. As such social navigation will be possible, whereby robots such as delivery robots will be capable of navigating busy streets without negative implications or impacting the comfort levels of humans [6]. Maintaining safe and sociable relations between humans and robots is, not only a key consideration in all robotic development, but an intrinsic factor that must be constantly considered when developing robot safety.

While safety organisations such as the ISO attempt to keep up with robotic development, the move away from human control makes the task of identifying safety necessities for a robot extremely difficult. Indeed, Vanderelst and Winfield even claim that autonomous robots should not just be safe but morally accountable and capable of making ethically sound decisions [7]. However, the implementation of ethical behaviours are hampered by a lack of versatility in robotic assessment of situations, as is discussed by the authors who point out that these ethical rules would not be suitable for every interaction. As such, it seems difficult to escape the need for human intervention to ensure even basic human safety around autonomous robots at present.

Another important aspect of human safety around autonomous robots is developing a set of criteria that can be defined as "safe" which can be implemented by the robot itself, rather than managed by a human. It appears that there is still extensive work to be done in moving the task of safety from the designer to the robot itself and humans will have an intrinsic role to play in this, in order to secure their own safety [8]. Moreover, solutions to this end need to complement robot safety and allow them to ensure their own self-preservation, as will be discussed in the next section.

3.1.2 Robot Safety

The criteria to ensure robot safety are similar to those used to ensure human safety, in that both require the robot to be able to react to new environments, make assessments of the obstacles and other humans/ robots they encounter and have sufficiently robust protocols to be able to react appropriately in these situations. However, ensuring robot safety goes a step further in that it requires the robot to be self-aware enough to wish to be safe and to act as such.

Robot safety can be compromised by several factors. Foremost among these is human interference, which can occur in many shapes and forms. As was discussed above, until robots can become part of the mesh of human society, they are often viewed with scepticism and in some cases fear. One study carried out by [9] investigated humans' reactions with service robots when the robots were left alone in public spaces. The findings of this study show that there was considerable human curiosity about the robot's capabilities, occasionally this curiosity degenerated into physical abuse of the robot, with people attempting to confuse or waylay it. There were also examples of more extreme cases, with groups of young people carrying out actions that were classed as bullying of the robot. The experiments carried out in this paper did not collect quantitative data but were conducted to examine how people generally react to unsupervised robots. As an experimental control, a supervised robot was also included in the experiment. This robot experienced no bullying behaviour from humans. As such, it is not difficult to imagine that as autonomous robots are increasingly integrated into society, they will face prejudice from humans, which could take the form of activities harmful to the robot.

It is worth considering that the threats posed to robots could come in many forms with the most likely danger being posed by humans. While humans might not always physically perpetrate an attack, human-controlled drones and hacks are a very likely reality of an increased number of autonomous and unsupervised robots in the world. Such possibilities are discussed in Risk-based Triggering of Bio-inspired Self-Preservation to Protect Robots from Threats [1] which suggests that by allowing robots to react differently upon the approach of a threat, increases their chances of remaining safe. The risk-based triggering technique relies on the fact that designers would have included sensing and real time processing in the robot which would enable the robot to identify dangerous elements in their environment such as would be pursuers and to monitor their behaviour. This paper examines a wide range of threats that could be experienced by delivery robots as part of their day-to-day work. The tests used in this paper, while preliminary, are well researched and lead to the conclusion that robots with inbuilt risk-based triggering fare better in conflict situations. This conclusion, along with the results presented, can be taken as informed evidence that self-awareness is one of the first steps towards ensuring robot safety.

Another aspect of robot safety that must be considered is their ability to monitor the vital parts of their operating systems. Failure to do so would result in robots being incapable of acting in a fully autonomous manner. Such a consideration has been explored by Castro-Gonzales et al through the use of reinforced learning [10]. They use the example of a domestic robot that must learn to go to the charger once it is running low on battery. They conclude that by allowing a robot to learn from its previous mistakes and thus building up its understanding and knowledge of a situation, increases its chances of self-preservation. This is also an important biological trait in animals and thus was incorporated into this project.

Robot safety is still in the early stages of development and this project has contributed to its continued growth.

3.1.3 Conclusion

Evidently, investigations into robot safety have focused largely on the benefits to humans, as opposed to the robots themselves. While this was appropriate before the emergence of autonomous robotS, decreased human control increases the need for robots to be capable of ensuring their own safety. Thus far, work into equipping robotS with self-preservation techniques has been largely fruitful. Further work into this will be beneficial to the robotics community. Once proper practices have been put in place for self-preservation techniques they should be used in conjunction with techniques such as reinforced learning. This combination will allow robotic safety to advance considerably.

3.2 Bio-inspired self-preservation and predator-prey relationships

While investigations into bio-inspired self-preservation behaviours for robots have not been extensive, there is extremely useful groundwork in the self-preservation of robots which was used as the foundation of this project. This section examines four papers that deal with bio-inspired self-preservation techniques. The particular techniques being investigated are risk assessment and reaction to threat.

3.2.1 Risk assessment

Previously examined self-preservation techniques are of the utmost importance to this project and will here be assessed. While the reaction to threat will be discussed below, a review of risk analysis in robots and their ability to dynamically assess the situation is firstly presented. These two stages of reacting to threat form the basis of predator evasion in the wild and were incorporated into this project.

In order to be aware of the risk associated with each situation, the first step is establishing a means of categorizing pursuers. This can be done in several ways. Chiu, Araiza-Illan, and Eder did this by initially establishing whether a pursuer was low risk, moderate risk, high risk or extreme risk [1] - HOW DID THEY PUT POTENTIAL PURSUERS INTO THESE CATEGORIES, WHAT CRITERIA WERE USED IN THE CATEGORISATION. In conjunction with each level of risk, an appropriate response was recommended for the robot. While this method does not allow for robotic development following continuous exposure to the same pursuer as the robot does not maintain information on the pursuer from one simulation to the next, it is effective in allowing the robot to classify the level of danger each pursuer represents. this method was adopted as the basis for the risk analysis aspect of this project. However, in order to build upon this technique, other self preservation variables were examined when calculating risk, for example the amount of time the robot is allowed to monitor the pursuing drone.

One of the most critical aspects of risk assessment centres on the robot's ability to assess whether or not the approaching threat is of more consequence than failure to complete its mission. As with animals in similar scenarios, it is not acceptable for self-preservation to be the only concern of a robot. A robot's ability to analyze threat while giving proper consideration to its other goals is discussed in [11]. In this paper the necessity of a robot to balance threat avoidance with other requirements (in this case feeding) is illustrated clearly. In one case the study examined the "Hiding Robot". In some simulations this robot was programmed with a low hunger threshold (feeding is a much lower priority than evading the predator), which resulted in the robot starving to death in almost every situation, such is their level of caution when facing a predator. Conversely, when the robot's priorities were reversed, the robot was captured while feeding almost every time due to the robot's high hunger threshold. The extremes of these two scenarios, where the robot was given a single priority that outweighed all other tasks demonstrates the need for robots to be capable of accurately assessing threats and their associated level of risk in order to work efficiently. While there are shortcomings in this study such as the environmental limitations which somewhat curtail the applicability of these findings, the authors' conclusion "having separate subsystems to deal with dangerous situations allows an agent to be less obstructed in undertaking its primary activities" is certainly backed up by the data collected.

In order for a robot to be capable of escaping a mal-intentioned pursuer, it is necessary for them to be aware of a threat being posed to them. Moreover, it is necessary for them to be capable of accurately assessing the level of risk that a threat poses and weighing this threat up against their other priorities.

3.2.2 Reaction to threat

Once a threat has been detected by a robot, enacting a well reasoned response to this threat is critical to robot safety. One aspect of this is intelligent threat avoidance, while the other is continuing to monitor the threat in order to confirm that the reaction takes no more time than is absolutely necessary.

Once a chase has begun, it is necessary for a robot to dynamically assess the situation, in order to have the greatest chance of avoiding capture. Two papers examine the best method for a robot to do this [12, 13]. Araiza-Illan proposes that the best way for a robot to avoid capture is to move erratically. This can be achieved by allowing the robot to set sub-goals [12]. These goals result in random, unpredictable direction changes which are similar to an animal in the wild, buying the prey time as they assess the situation and seek a refuge. This idea of creating random movement in a robot is also explored by Curiac and Volosencu who programmed a patrol robot's sensors to detect fictive, temporary obstacles, which, in moving to avoid them, caused the robot to move erratically. They consider the fact that by causing only temporary variations in the robot's path, the robot retains its main goal throughout [13]. Essentially the two methods being examined are the same idea, implemented through different strategies. Both methods cause the robot to move unpredictably, increasing the distance between them and their pursuer and allowing them time to assess their situation. While the implementation of one of these strategies would have been ideal for this project, limited time allowed only for the attachment of a sensor that monitored the predator's proximity. While this was somewhat crude, it allowed the robot to continuously dynamically assess the level of risk associated with the threat it was facing. A robot's ability to reassess their levels of danger is necessary for implementing bio-inspired behaviours in a robot. It allows the robot to adapt to the changing conditions of the chase while it is in motion and to continuously dynamically assess their level of risk.

In order for a robot to carry out an intelligent reaction to a threat, it is necessary for the robot to balance immediate threat avoidance with other self-preservation considerations. Other self-preservation considerations include battery power, without which the robot would be unable to return to its base. As such, when reacting to a threat it is necessary for both short and long term self-preservation considerations to be taken into account. Failure to do can result in low success rates as can be seen by the varying success rates of the four robots in [11]. A comparison of the "Hiding Bot" - which is discussed above section 3.2.1 - compared with any of its counterparts demonstrates that evasive action is necessary to robot survival in these scenarios. This is similarly demonstrated in [1] with robots who are programmed with a risk-based mechanism having a far higher chance of survival as they can respond to the presence of a predator. Creating an intelligent response that balances predator avoidance with other self-preservation considerations (e.g. battery recharging) is similarly important. An example of how this can affect a robot's success can be seen in a comparison of the "Running Bot" and the "Memory Bot". The "Running Bot" has a risk avoidance mechanism built in which allows it to run to a refuge when it sees the predator. While this approach is more successful than the "Hiding Bot", the "Running Bot" only ventures out of the refuge when hungry. This means that all of its encounters with the predator are high-risk i.e. if it is intercepted before it feeds it has the possibility of dying of starvation. This is far less likely with the "Memory Bot" who does not wait until hunger levels are low to venture out of the refuge. The "Memory Bot" remembers when the last time it saw the predator was and based on this information, allows a reasonable amount of time to lapse before venturing out of the hiding place. The "Memory Bot" bases its emergence from the hiding spot based on time and not based on its hunger levels and has a higher level of success than the "Running Bot". As such, it is clear that a robot's reaction to a threat must immediately take into consideration a number of both long-term and short-term self-preservation considerations.

3.2.3 Conclusion

The works discussed show the importance of implementing self-preservation techniques and intelligent reaction to threat in robots. In all cases, the inclusion of these behaviours improved a robot's chance of avoiding capture or surviving. There appears to be particular benefit in a robot being capable of dynamically assessing its level of threat, without neglecting more long term self-preservation considerations. Based on the findings above, risk-based triggering was considered central to any assessment carried out on robotic self-preservation and was a focal point of this project.

3.3 Animal behaviours in predator-prey relationships

In this section it will become clear where the bio-inspiration for this project comes from. It will be necessary to examine a wide range of predator-prey relationships from which inspiration can be drawn for similar situations involving robots. The focus will mainly be on land-based animals as the robot under investigation is also land-based, however, some reference to other animals will be included if appropriate. The sets of behaviours being monitored in animals has been split into two sections;

- flight initiation distance
- refuging

Both of these behaviours will be examined closely in their embodiment in the animal world.

3.3.1 Flight initiation distance

Flight initiation distance is of utmost importance in natural predator-prey relationships. Flight initiation distance is the distance from the predator at which the prey chooses to flee. Probably the most important aspect of calculating flight initiation distance is the calculation of the cost of remaining versus the cost of fleeing. This is a calculation of the risks of remaining i.e. being captured versus the risks of fleeing e.g. missed mating and feeding opportunities and is heavily reliant on predator proximity. This section will discuss the impact that the cost of fleeing and predator proximity have on animals' flight initiation distances.

3.3.1.1 Cost of fleeing The underlying factor taken into account in any predator-prey relationship when calculating flight initiation distance is whether the cost of fleeing outweighs the cost of staying. In papers written by both Stankowich and Coss and Stankowich and Blumstein there is discussion as to how it is unbeneficial for the prey to flee immediately upon seeing a predator unless they fail to notice the predator until they are very close. As such, the prey must weigh up the benefits of staying against the benefits of leaving. Territorial animals, animals that are feeding, mating or guarding young tend to allow for smaller flight initiation distances [14]. This is because the result of fleeing will be a loss of these opportunities or, in the case of guarding young, may result in the young falling behind and being hunted. As such, the most beneficial self-preservation technique is for an animal to remain where they are until the chances of them being attacked increase to a critical point. At this point, the cost of staying is said to outweigh the cost of fleeing. In contrast, most non-feeding or non-territorial animals tend to allow for greater flight initiation distances as the cost of fleeing is relatively low [14]. While both papers examine predators that approach on land as opposed to from the sky, their assessment of the calculation of the cost of remaining versus the cost of fleeing is no less applicable to delivery robots. In the case of a delivery robot, the cost of fleeing is not completing their mission or completing it inefficiently. It is critical that the decision to flee is not taken too early by the robot in order to ensure maximum performance. Conversely, robots who are particularly vulnerable to threat are of little value to the companies who own them. Weighing up the cost of fleeing against the cost of staying or remaining on mission is an intrinsic part of flight initiation distance calculations for delivery robots and forms a central part of this project.

3.3.1.2 Predator Proximity An important factor that is generally considered by animals when calculating the cost of fleeing is the proximity of the predator. If the predator's chasing velocity is high, the chance of the prey outrunning the predator is slim. Stankowich and Coss explore a simple method for measuring how animals determine their flight initiation distance by considering there to be a maximum distance (Dmax) and a minimum distance (Dmin) which represent the limits of their surveillance time [15]. They assert that if a perceived predator is outside of Dmax , regardless of their behaviour, the prey will ignore their movements and maintain a relaxed scanning rate. Once the predator is between Dmax and Dmin, the prey will become alert, monitor its movements and increase its scanning rate [14]. If the predator reaches Dmin, then the prey will flee as they will perceive remaining as being too dangerous. Moreover, if the prey for some reason is unaware of a predator's approach and only becomes aware of it at a very close distance, they will instinctively flee as they will not have time to determine a suitable Dmin or assess if the predator has gone beyond that distance yet [15]. Thus, Dmin can be treated as the animal's flight initiation distance. At a closer distance than Dmin the cost of fleeing becomes irrelevant as capture is the most likely outcome. In a robot-drone scenario, Dmin would be the optimum distance the robot would allow the drone to approach before deciding to abandon mission in favour of survival. If the optimum is found, this would allow the robot to handle predatory encounters in the most efficient way possible. Predator proximity is a key feature assessed both initially and dynamically throughout the course of these experiments.

3.3.2 Refuging

Once the prey has begun its flight, it has two potential courses of action; it can either flee to a safe distance or it can find a refuge. Because it has been established that in the case of a delivery robot-drone relationship simply fleeing is not the ideal behaviour [1] refuging will be explored in this section with consideration given to how animals use refugia in their attempts to avoid capture in predator-prey interactions.

In their study examining self-defense strategies in delivery robots, Chiu, Araiza-Illan, and Eder found that finding a refuge, combined with fleeing, was a good self-preservation policy for robots when being pursued [1]. It is a trait shared by many animals who tend to flee to refugia where possible in order to escape a predator. Thus, while it is assumed that fleeing to a refuge will be an intrinsic part of this project, the amount of time the robot should remain in the refuge, similar to animals in the same situation, is of critical importance. While a refuge is considered safe, it is not possible for an animal to remain in a refuge indefinitely. For example, in their discussion of the case of lizards, Cooper et. al point out that lizard's body temperatures often drop while they remain unmoving in an often-shaded refuge, to the detriment of their basking time. Lizards rely on sun-basking to ensure their body temperatures remain constant and healthy, thus practices that prevent this are harmful to the lizard [16]. In such cases, the time spent in the refuge tends to be short whereas, for lizards who have achieved a higher body temperature, time spent in a refuge can be considerably longer [17]. The examinations carried out on refuging lizards, while somewhat haphazard in style, given that the predator was a human, serve to confirm the fact that a refuging animal cannot stay indefinitely in a refuge. This theory is particularly transferable to robots, who similar to lizards, require recharging of sorts. The robot's battery levels can be seen as similar to a lizard's body temperature as it cannot complete its mission if its battery level drops to zero. Moreover, unlike a lizard who must only leave a refuge in order to recharge, a robot must leave sufficient battery power to reach a recharging point in order for its time in refuge to be worthwhile. As such, a critical aspect of biological predator-prey relationships that will be studied in this project, is establishing an optimum refuging time.

A factor proposed by Martin and Lopez that deeply affects the time spent in refuge, is the perceived risk level of the attack [17]. If the predator displayed signs of aggressive intent and pursued the prey in earnest it would be considered a high-risk attack. This would result in a longer time being spent in the refuge to increase the chances of the attacker having withdrawn by the time the prey emerges [17]. This is a critical aspect of an attack that a robot must take into account as it can have a substantial impact on the efficiency and success rate of a mission. By ensuring that a robot can accurately assess the risk level of an attack, it is possible to mitigate against them remaining in a refuge longer than is necessary. In the case of this project, this was taken into account by assessing the proximity of the drone and its persistence outside the refuge. As is the case with animals, in programming the robot to take into account the risk level of an attack, a more efficient refuging time can be calculated which in turn allows for an efficient completion of the robot's mission.

Refuging comprises a crucial aspect of successful responses to predators in biological predator-prey relationships. While refuging, animals must continuously assess the benefit of remaining in the refuge and make intelligent decisions about the length of time required to avoid the predator. This ensures that refuging remains beneficial to animals and does not result in diminished energy levels or survival prospects. As such, this is an element of biological predator-prey relationships that must be included in robotic predator-prey relationships and establishing an optimum, gives the robot the best chance of operating efficiently and successfully.

3.3.3 Conclusion

Each of the behaviours discussed in this section are of benefit to animals who are the prey in predator-prey relationships in the wild. Those behaviours are:

- flight initiation distance and
- refuging

Their ability to combine these behaviours in order to establish whether a situation is high-risk or low-risk allows them to develop the most efficient and effective escape strategy. This is the bio-inspiration that was taken for this project. The tests carried out incorporated, as best as possible, these behaviours in an attempt to allow delivery robots to carry out similar assessment protocols. While animals must weigh predatory risks against necessities such as feeding and eating, the delivery robot weighs predatory risks against the necessity of reaching its goal efficiently and thus successfully completing its mission. The measure of success for this project takes both the robot's ability to avoid capture and its ability to reach its goal efficiently into account. Therefore, it is possible for a robot to embody bio-inspired behaviours as its sole focus in escaping capture.

3.4 Summary

This literature review has provided a thorough investigation into the three key areas associated with this project. Robot safety is an emerging field that has been largely overshadowed by human safety. It is in need of extensive work in order for autonomous robots to become a staple of everyday life. Previous investigations have begun looking into self-preservation in robots and have provided exceptional work that can be used as the basis for this study. However, there is yet to be a thorough examination of bio-inspired behaviour and their effect on a robot's ability to avoid capture. This project hopes to fill this gap in robotics study and present a useful conclusion that can be developed and carried into the future.

4 Project Planning

The parameters of this project, and the thought process behind their selection, are discussed in this section. Despite working in a simulated environment, the parameters set out for this project were strictly enforced in order to assure its usefulness in real world application. Each scenario tested was based on the specific requirements of robots carrying out different types of missions, each with a distinct set of priorities. Figure 1 shows both of the robots, their respective set of priorities and how these priorities were simulated.

Type of Robot	Cautious	Reckless
Priorities	Self-preservation: -Survive encounter	Efficiency: - Conserve energy - Complete task quickly
Reactions	- 30m flight initiation distance - Remain in refuge until drone leaves	- 5m flight initiation distance - Remain in refuge for 30 seconds

Figure 1: Table displaying different types of robot and their respective characteristics

4.1 Setup Process

As can be seen in Figure 1 there are two varieties of the same robot distinguished by their set of priorities. As a result of these priorities, the robots reactions have been altered to fit with their priorities. A thorough explanation of each of the columns is given below.

4.1.1 Priorities

Robots are generally programmed to complete their task first and foremost, making this their top priority. However, in this experiment, other factors have been taken into consideration such as battery consumption, speed and survival.

- Battery Consumption - One of the priorities for the robots was battery consumption. The robot who would be required to carry out more than one task at a time before returning to base was programmed to be less cautious, spending less time hiding and running from the drone.
- Speed - Speed was tied in with battery consumption in that the robot who needed to complete several tasks at a time was also expected to be fast. As with battery consumption, the robot that was programmed to prioritise speed was less cautious.
- Survival - The robot that had less restrictions on battery usage and speed was programmed to prioritise survival. It was programmed to be more cautious with less restriction on how long it could hide for and more battery power dedicated to outrunning the drone.

4.1.2 Type of Robot

The robots included in these experiments were varied in terms of the level of their caution when faced with predators. In order to differentiate between the robots one is considered the "Cautious Robot" and one the "Reckless Robot".

4.1.3 Reactions

The reaction that individual robots have to the approach of the drone is based on their set of priorities. The reactions are intended to mimic those of animals in predator-prey-relationships who find themselves in similar situations. Two specific aspects of the robots reaction were altered based on the priority set assigned to the robot.

- Flight Initiation Distance – As defined in section 1 the flight initiation distance of an animal in a predator prey relationship is the distance from the predator at which they choose to flee from their position. For the robots, flight initiation distance is the distance from the drone at which they abandon the mission and choose instead to seek refuge.
- Refuge Time – Because the ideal predator avoidance mechanism for a robot incorporates fleeing and hiding, it was necessary to dictate how long each robot was allowed to remain in refuge. While in the refuge the drone considered the robot uncatchable.

4.1.4 Conclusion

In order to carry out this project in such a way that it could be easily replicated, it was necessary to map out a clear set of parameters for each robot. The priorities of the robots are based on the expected priorities of the robot designers in real world applications and the reactions are biologically inspired, drawing inspiration from predator-prey relationships in the wild.

4.2 Project Assumptions and Limitations

In order to ensure the timely commencement of this project, it was necessary to consider certain elements of robot self-defence strategies as proven to be true. There are also a number of limitations associated with the project, largely due to time and financial restraints. Both the assumptions and the limitations of this project are outlined below.

4.2.1 Assumptions

Many key factors at the foundation of this project were considered true based on prior research carried out by Computer Science and Biology researchers.

- Prey Avoidance of Predatory Attacks – for the sake of this project, it was assumed that prey who choose to avoid an encounter with their predator are more likely to survive. This has been proven several times over in papers such as CITE. Due to the nature of the robots being used in this experiment it was assumed that the best animals to study would be ground-based ones. Moreover, it was assumed that the drone attacks would mimic those of a predator in the wild to the extent that they could be called "predatory attacks". This was as opposed to a more erratic or destructive form of attack.

- Necessity of Risk-Based Triggering Mechanism – As it was proven in [1] that robots who were programmed with risk-based triggering mechanism were more successful at avoiding predators than those without, it was assumed that using a risk-based triggering mechanism would aid in programming the robots' self-defence mechanism. The risk-based triggering in this experiment is seen as the variety of sensors that the robot is programmed with that allow for tracking of the drone as it approaches.
- Self-Defence Strategy – The results of the experiments carried out in [1] suggest that the most effective way for a ground-based robot to avoid being captured by a drone is to combine refuging and fleeing in their self-defence strategy. As such, all robots were programmed to incorporate a both elements as part of their evasive reaction.

4.2.2 Limitations

While this project has explored its thesis to the fullest extent possible, the project could be improved upon due to the existence of a number of limitations associated with it. Some of these limitations are outlined below.

- Simulation Environment – The limitation that had the greatest impact on this project was the fact that it was conducted in a simulated environment. The effects of this are twofold. Firstly, while the conclusions drawn in this thesis are based on the results available they would not necessarily be the same as if the project had been conducted with real robots. Secondly, the environment itself has certain shortcomings that make it dissimilar to the real world. For example, the simulation of gravity is not ideal with certain glitches occurring from time to time. In cases where this had a visible effect on the results of the project, the data was discounted, however that does not negate the fact the the results could have been effected by this. Similarly, because this is a simulated environment the robot and drone have access to each other's locations. While the robot does not make use of this, the drone's tracking is unrealistically accurate due to the fact that it is reading the robot's location and moving towards this. As such, while it was the best available option, using a simulated environment included certain shortcomings.
- Time Constraints – Due to a limited amount of time in which the project could

be completed, certain elements of the project were shortened. As mentioned above, the drone's tracking system is based on reading the robot's location in the same way that the robot reads it. While this is unrealistic, it was an effective way of recreating a mal-intentioned drone while still being able to dedicate the majority of project time to working on the robot's self-defence mechanism. Similarly, implementing effective path planning algorithms in the robot would have been an excellent addition to the project and increased the reality of the siulations and the robot's efficiency. However, this was unfortunately not possible due to the limited time allowed for this project.

4.2.3 Summary

The experiments carried out over the course of this project were based on a strict set of parameters decided upon based on prior research. In order to carry out these experiments in a timely manner, certain assumptions were made and used as the basis of this project. While the project was run effectively within various constraints, there were limitations associated with it that may would suggest that further research is necessary to conclusively determine the impact that flight initiation distance and refuging time have on a robot's success rate and efficiency.

4.3 Experiment Design

In order to accurately address the question of how flight initiation distance and refuging time impact a robot's success rate and efficiency, it was necessary to carefully design the experiments. In order to do this, use was made of Morse simulation software, which was considered to be the best way to simulate the required environment. All robot programming was in Python. Once the simulation software had been decided upon, it was necessary to draw up strict specifications as to the environment design, the robot design and features and the software design. Each of these features is discussed in detail below.

4.3.1 Technical Background – Morse

Morse simulation software is an open source generic simulator for academic robots, as defined in section 1. It was ideal for this project as it provided a number of pre-built features. The project makes use of the Morse "Outdoors" world which was considered the most appropriate for the experiments being carried out as it was the closest to the expected environment of a delivery robot. Similarly, the drone and robot models were both included in the Morse package which meant that no robot design was necessary and it was possible to focus on programming robot behaviour. Finally, a number of the sensors and components built into Morse were included in the project at one stage or another. While there has been an attempt to reduce the number of sensors included in the final stages of the project, throughout the course of the experiment design a large number were included for a variety of reasons e.g. the orientation sensor was briefly considered to be useful for collision avoidance, however this was quickly revised.

4.3.2 Environment Design

As mentioned in section 4.3.1 there are a number of simulated environments available in Morse, therefore it was necessary to settle on the features that were required for the projects to be run. Each of these stipulations are outlined below.

- Clear End Point – Central to the experiments being carried out was the existence of a clear end point. In order to simulate a lifelike mission, it was necessary to have a definite end point towards which the robot could move. Moreover, the robot had to be programmed to recognise the fact that it had arrived and that its mission was complete in order to end the simulation.
- Traversable Land – In order for the robot to be able to move easily through the scene, it was deemed necessary that the ground be made up of an an easily traversable substance. Ideally, concrete would have been used as this is the most realistic surface on which a delivery robot would be moving. However, as concrete was not available it was decided that grass was the next best option and, as such, the "Outdoors" theme was selected.
- Refuges and Obstacles – In order to assess the robot's ability to seek refuge and avoid obstacles, it was necessary that the selected environment should include obstacles such as trees, buildings, long grass and rocks. Initially it had been hoped that the robot would be capable of using the trees included as a refuge, however, the "Waypoint" collision avoidance was not sophisticated enough to maneuver around complex objects, therefore it was decided the hiding points would be among long grass which was marked as a safe zone for the robot. By selecting an environment that included refuges and obstacles it became possible to add reality to the simulation.

Bearing these requirements in mind, the "Outdoors" environment was selected as the most appropriate. Its inclusion of both open spaces and buildings made it easy for the experiment to be monitored by eye and added to the reality of the scenario. The air space was varied enough that the drone sometimes had clear flight paths while other times was obstructed by trees. It also included a range of flat ground and inclines which, similarly, made the simulations more realistic.

4.3.3 Robot Design and Features

A crucial element of the experiment was the robot design. The key factors taken into consideration when deciding upon this were

- the physical traits required in the robot most similar to a delivery robot and
- the sensors needed for the simulations to run

both of which are discussed below.

- Physical Traits - It was decided that the robot should be ground based in order to best represent a delivery robot. The robot had to keep generally low speeds as this would be required of any robot operating in a delivery capacity. This results in the drone being capable of running down the delivery robot, operating at 2.5 times the robot's normal speed.

Figure 2: ATRV used in experiments.

- Sensors - The robot had to be capable of monitoring the drone's proximity in a realistic way. This was done by using the "Proximity Sensor" which monitors any object that enters a thirty meter radius around the drone. It classifies all objects that enter that zone and as such could recognise the drone. As such, the robot used the "Proximity Sensor" in much the same way an animal uses its senses to detect threats. Similarly the robot had to be capable of navigation in order to programmed with a goal. This was achieved using the "Pose Sensor" and the "Motion Sensor". The "Pose Sensor" allowed the robot to keep track of its current position while the "Motion Sensor"ensured it was possible to program goals for the robot. Finally the "Battery Sensor" was attached to the robot which monitored the robot's battery levels. This was critical in producing the required results of this project as it allowed the robot to output its battery level upon the conclusion of each encounter.

Strict requirements for the robot's design ensured that the experiments conducted in this project were as realistic and informative as possible. The robot was physically similar to a model of a delivery robot and was equipped with sensors that allowed it to carry out the desired simulations, without enhancing its abilities beyond the constraints of reality.

4.3.4 Software Design

The software design of this project was relatively simple. There were four different scripts that needed to be written:

- Environmental Control Script - This script created the environment and the components within it. It was within this script that the starting points for the drone and the robot and sensors to be included on the drone and the robot were specified.
- Drone Control Script - This script was initially based on the Morse tutorial "Flying Cat and Mouse Game" which produces a drone who follows a keyboard-controlled robot. This basis was built upon extensively, allowing for the inclusion of the "Battery Sensor", capture abilities, robot safe zones and feedback to the terminal.
- Robot Control Script - This script controlled the robot. Within this script it was possible to control the robot's responses and to assign it hiding places. This is the script which varies between the different versions of the robot depending on the level of caution being exercised. This was also the script responsible for outputting the experimental results to the terminal.
- Bash Script - This script controlled the other three, running them consecutively at specific intervals. It was the script responsible for outputting results to a CSV and for generating the random head start the robot had over the drone.

While the implementation of many of the desired traits took a lot of time to execute, the design of the software controlling the experiments in this project was relatively simple. The information and examples available on the OpenRobots (Morse) website proved particularly useful in providing code snippets and solutions.

4.3.5 Conclusion

The factors that were taken into consideration for these experiments allowed them to be conducted in a manner as close to life as possible. The standards set for these experiments ensured that the robot being tested was as similar to a delivery robot as possible. The use of Morse provided an excellent simulation environment and the components included in it could easily be adapted to fit with the specifications of this project. The code design was relatively simple but worked effectively to accomplish the goals set out for the project. The planning of this project was extensive but a useful investment of time as it ensured a high standard for all tests conducted.

5 Project Execution and Results

This section will detail how the specifications detailed in section 4.3 were implemented. It will outline and discuss the results collected from the carrying out of these experiments and will attempt to highlight the main achievements of this project.

5.1 Methodology

As was discussed in section 4.1.2 there were a number of scenarios being tested in the experiments throughout the course of this project. Each scenario was simulated under the same conditions following a strict methodology.

This section will outline the various different stages of the methodology and will highlight differences between the individual scenarios. Specifically, this section will discuss the following:

- Initialising the Simulation
- Robot and Drone Starting Points
- Randomized Head Start
- Programming the Reactions
- Conclusion of Simulation
- Running the Simulation
- Recording Data

5.1.1 Initialising the simulation

Due to the fact that this experiment was run via simulation, for each iteration of the experiments, it was necessary to initialise the simulation. In order to monitor the projects by eye, the GUI included in the Morse framework was used. This added to the time taken for initialisation of the scene but once initialised had little effect on the timeliness or efficiency of the project.

This experiment had three controlling scripts; one for the environment, one to control drone movement and one to control robot movement. Due to the fact that there were three scripts running simultaneously for the simulations in the project it was necessary to run a Bash script which would be capable of ensuring they were run at constant intervals for every simulation carried out. The first script run was the environmental control script. In order to begin a simulation in Morse it is necessary to use the "morse run" command which creates the environment and places the components and cameras in their required places. This process is resource intensive and as such involves a substantial time delay. As a consequence there was a twenty second time delay between the command to initialise the environment and the command to begin the robot actions.

Figure 3: Screenshot of environmental setup in initialised scene.

The scene was initialised such that the camera was located in one of the corners of the environmnet, directly behind the robot. From this position it was possible to see most of the simulation unfold, despite obstacles within the scene. A second camera was included in the scene which was attached to the drone. This allowed for observation of the drone's movement and line of vision. This camera was displayed in the top right hand corner of the scene, such that it was possible to observe it alongside the main scene.

Once the scene was intialised in this way it was possible to begin the robot control script.

5.1.2 Robot and Drone Starting Points

Within the simulation, the robot and drone were given specific starting points in order to allow as much time as possible for the simulation to take place. Due to the limited size of the environment and the fact that it was decided the robot should conclude its mission by arriving at one of the buildings, it was necessary to position the robot such that its journey would take enough time that it would not have reached its destination by the time the drone reached it.

The robot was placed in the corner of the world furthest from its delivery destination. Moreover, its starting point allowed it to spend the first section of its journey moving along a path included in the world. This was assumed to be closer to a concrete surface than grass and as closer to life. Unimpeded, it took the robot roughly one minute to reach its destination and throughout the process it was possible to observe the entire mission from the main camera.

The drone's starting position needed to be in such a location that it allowed the drone to intercept the robot in the midst of its mission. As a result, the drone was placed roughly midway between the robot and its destination, to the far left of the environment. This ensured that the drone would either move between the robot and its destination or follow the robot towards the destination, depending on the time delay. This ensured that the robots responses while either close to its starting point or close to its destination could be tested. Moreover, the drone's starting point included a number of trees between the drone and the main path on which the robot moved. As such, the drone's flight could be thrown off course if it collided with items such as leaves or branches. This added to the reality and randomization of the scene as the the drone being knocked off course resulted in the time ff the robots flight initiation being similarly altered.

5.1.3 Randomised Head Start

In order to introduce a randomised element into the experiments carried out in this project, it was decided that the robot would be given a randomised head start on the drone of between one and thirty seconds. Due to the length of time it takes the robot to reach its destination from its starting point, it was decided that thirty seconds represented a fair limit for the length of this delay. If the delay was more than thirty seconds, the drone's impact on the experiment would have been minimal, particularly seeing as the last fifteen second of the simulation see the robot protected by a large building which oftentimes prevented the drone from aproaching too close. While a longer delay may have been more realistic, it would not have helped to address the hypothesis at the centre of this project. This delay was implemented in the Bash script and manifested itself as a delay between the launch of the robot control script and the drone control script. While this random element initially seemed conservative, it transpired that on many occasions different results were collected in different occurrences of the same time delay as can be seen in A Appendix. As such, it was concluded that, while the delay was the only specifically randomised feature of the experiment, as discussed above section 5.1.2, the collisions the drone encountered over the course of its flight acted as a second random feature. By including a randomised head start for the drone, it was possible to test a wide range of scenarios and gather a more comprehensive set of results.

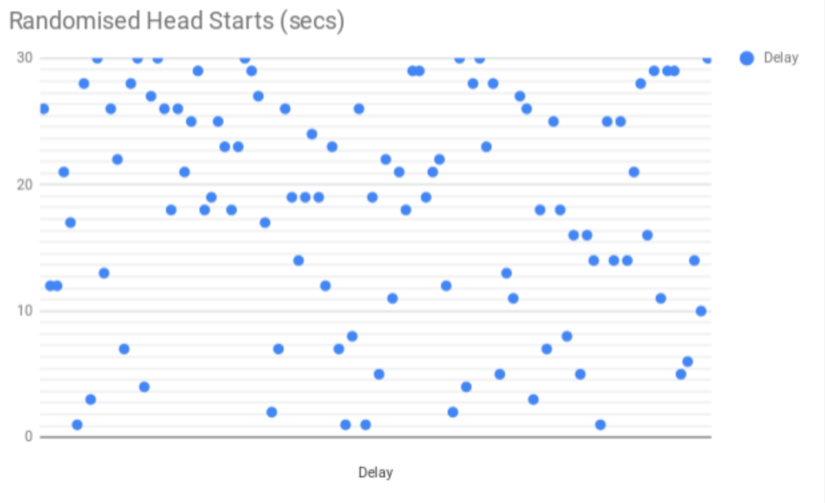

Figure 4: Randomised head start in seconds for Reckless robot simulations

5.1.4 Programming the Reactions

Possibly the most important aspect of the experiments was the programming of the robot's behaviours. This was central to the investigation and allowed robot's with different reactions to be compared in terms of their mission success and efficiency. Two different versions of the same robot were tested on; one that was extremely cautious and one that was extremely reckless. The individual behaviours assigned to each will be discussed below.

In both cases the robot's behaviour was modified in the control script for the robot. The behaviour was expressed as the distance - as detected on the "Proximity Sensor" - the robot should allow the drone to get before moving to a refuge and the amount of time the robot should remain in the refuge once reached. The robot was programmed to move to the nearest of the three hiding places once it had elected to refuge. The proximity of the hiding place was based on the Euclidean distance between the robot's current position and the fixed hiding places.

5.1.4.1 Cautious Robot The cautious version of the robot was programmed to flee as soon as the drone was detected on the "Proximity Sensor" i.e. within thirty meters. Once the decision was taken to flee, the robot's speed increased to double its original speed and it moved towards the refuge. If it reached the refuge, it remained there until as long as the drone remained within the thirty meter limit i.e. while the drone's battery level was more than fifty percent. Once the drone returned to its base, the robot emerged from the refuge and continued towards its destination.

5.1.4.2 Reckless Robot The reckless version of the robot had the same detection on its "Proximity Sensor" and detected the drone once it came within thirty meters. At this point it picked up its speed but did not stop its mission. The reckless robot only left its mission to flee when the drone came within five meters of it. At this point it would move to the nearest refuge. If it reached the refuge, it would only remain there for t seconds before leaving the refuge and attempting to outrun the drone to its destination.

By programming these two extreme cases it was possible to assess the impact the flight initiation distance and refuge time have on the speed and efficiency of a delivery robot. The behaviours were varied enough as to allow distinct differences to become apparent in the data collected from the simulations. The distances and times were based on estimates made based on the research that had been done into animal behaviours and as such are bio-inspired.

5.1.5 Conclusion of Simulation

There are two possible ways in which the simulations carried out in this project could end; either the robot was captured or the robot would reach its destination. In cases where the robot was captured the entire simulation stopped. In cases where the robot reached its destination the simulation would continue but the robot and drone's control programs would stop. At this point it was possible to manually shut down the simulation. In some cases, due to issues with collision avoidance, it was necessary for the program to be shut down manually if either the drone or the robot got stuck. In these cases, the data was disregarded as it did not add to information on either the drone's success rate or efficiency.

5.1.6 Running the Simulation

The process of running simulations for each variation of the robot was as is detailed above. Simulations for each version of the robot were carried out one hundred times each, with the required results and delay recorded for each.

5.1.7 Recording Data

Critical to the use of this project in future robotic work was the collection of accurate and useful data. There were four types of data recorded after each simulation had been run; whether or not the robot was captured, whether or not the robot attempted to hide, how long it took the robot to either reach its destination or be captured and the robot's battery level when it either reached its destination or was captured. The latter two pieces of information were outputted by the robot during the simulation. The Bash script was used to transcribe this data into a CSV as discussed above section 4.3.4. The robot kept track of the simulation running time and its battery level throughout each simulation and noted the fact that it was hiding from the point that it made the decision to hide until it had left its refuge again. As such it was possible to see if the robot had attempted to hide, even if the entire simulation had not been watched. All data was stored in CSV files in local and remote copies. A sample of the data collected for each of the two robots can be found in A Appendix.

5.1.8 Summary

The methodology outlined in this section ensured that the experiments in this project were carried out in a uniform manner. Moreover, the specifications which are outlined mean this project can be replicated by other researchers investigating similar topics. The environment was set up in such a way as to test the impact that flight initiation distance and refuge time had on robot success rate and efficiency based on many different approaches by the drone. The random factor introduced allowed a greater quantity of data to be collected for each robot and the drone's complex flight path added to this. Based on these factors along with the number of simulations carried out and the manner in which data was accurately collected the results discussed below can be assumed to be as accurate and whole as possible for an experiment of this nature.

5.2 Results

In the below section the results gathered in the experiments over the course of this project are laid out and discussed. The results presented show the four factors under investigation, they are:

- success rate
- refuging rate

- average time taken to complete simulation
- average battery level on completion of simulation.

These results are presented for both variations of the robot. Moreover, the rate of success versus the rate of failure is graphically represented. The results in their entirety are included in A Appendix.

5.2.1 Reckless Robot

The first robot tested was the reckless robot. This robot allowed the drone to approach to within five meters before fleeing to a refuge. Furthermore, the robot only remained in a refuge for thirty seconds once it had arrived and as such carried out a high risk strategy. The results of the one hundred simulations carried out using this robot are shown below.

	Reached Goal	Reached Hiding Place	Average time take	Average battery usage
Reckless Robot	4%	10%	39.9579 (secs)	2.60328 %

Figure 5: Results for Reckless robot

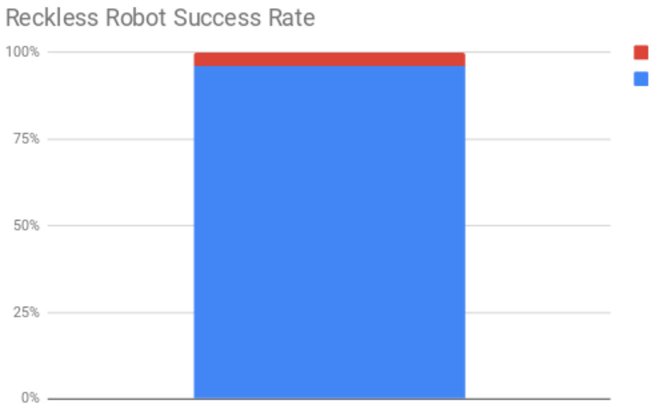

Figure 6: Success Rate for Reckless robot. Blue represents failures, red represents successes.

5.2.1.1 Observations The reckless robot almost never successfully reached its destination. As can be seen in 6 the success rate was only 4%. This rate is extremely low and suggests the robot would almost never complete its mission. Due to the high-risk strategy adopted for this robot, the time taken for the simulation is very short at an approximate average of 39.96 seconds, with the longest simulation lasting a total of 81.24 seconds. Conversely, the robot's refuging rate is high as it was almost never close enough to its destination to outrun the drone. As such, the robot came within five meters in almost every scenario and the robot was forced to flee to a refuge at a late stage in its mission. The robot's battery level at the end of each simulation remains high. This is due to the fact that, despite moving quickly through the simulation, it does not spend much battery power waiting in the refuge. As a result, in scenarios where the robot successfully completes its mission, it has ample battery to return to its base.

5.2.2 Cautious Robot

The second robot tested was the cautious robot, whose traits were in direct contrast to the reckless robot's. This robot fled immediately when its proximity sensor picked up the drone i.e. within thirty meters. Once the cautious robot reached its refuge it remained there until the drone ran out of battery and was forced to leave. The results of the simulations run on this version of the robot are presented below.

	Reached Goal	Reached Hiding Place	Average time take	Average battery usage
Cautious Robot	99.88%	99.92%	972.751 (secs)	49.2133 %

Figure 7: Results for Cautious robot

5.2.2.1 Observations The cautious robot is in direct contrast with with the reckless robot, almost never failing to reach its destination. Its success rate is 99.88%, an extremely high proportion of runs. Similarly, it reaches the refuge in 99.92% of simulations which has a direct hand in its success rate, considering once it reaches the refuge it is almost impossible for the robot to be captured. The scenarios where the cautious robot did not successfully complete its mission were almost all because the drone did not flee far enough, thus trapping the delivery robot in its refuge until it ran out of battery. The average time taken for the delivery robot to complete its mission is extremely high, as is its battery consumption. In almost all cases, the delivery robot completes its mission with less than fifty percent battery, meaning if it experienced similar delays on its return, it would not be capable of reaching its base. This demonstrates the inefficiency of the cautious robot's tactics.

5.3 Discussion

This section will discuss conclusions that can be drawn from the results collected over the course of this experiment. It will highlight patterns that emerged over the course of both simulations and the limitations of the experiments conducted over the course of the project which should be taken into account when considering the results of this project.

5.3.1 Conclusions

The results collected from the two variations of the robot differ greatly. The reckless robot almost never reaches its destination while the cautious robot almost never fails to reach its destination. The reckless robot's low success rate is largely due to the fact that it allows the drone to approach to a very close distance before electing to flee. Moreover, the robot does not remain in the refuge very long and instead relies on outrunning the robot to its destination. As such, it can be concluded that an extremely low flight initiation distance and short refuge time has a negative impact on a robot's success rate. However, in cases when the robot is successful, it completes its mission very efficiently. Thus, it can be concluded that short flight initiation distance and refuging time add to the robots efficient completion of missions.

The cautious robot maintains high flight initiation distances and refuging times. The results presented in 7 demonstrate that this leads to a high success rate. The cautious robot almost always reaches the refuge and remains there long enough that the drone has left by the time it emerges which suggests this is similarly positively impacted by high flight initiation distances and refuge times. However, the time it takes for the robot to complete its mission combined with the high rates of battery consumption demonstrated in the figures in 7 suggest that high flight initiation distances and refuge times negatively impact the efficiency of the robot. Moreover, in certain cases the robot remained trapped in the refuge due to the drone's proximity and in doing so ran down its battery without reaching its destination which was a useless waste of resources.

Overall it can be concluded that these two variations of the robot provide the edge cases for this study. The first sacrifices mission success in the hopes of increased efficiency while the second sacrifices efficiency for self-preservation. As such, it is evident that extremely low flight initiation distances and refuge times have a negative impact on mission success and extremely high flight initiation distances and refuge times have a negative impact on robot efficiency. As such, in order to find optimum flight initiation distance and refuge time for a robot in a predator-prey relationship it is necessary to find a middle ground between the two scenarios investigated in this project.

5.3.2 Limitations

The results and conclusions discussed above must be considered within the framework of the limitations of the project. These limitations are outlined below.

- Simulation - The conclusions drawn from the findings of this project are limited by the fact that the experiments were carried out in simulation. While the Morse simulator was extremely good, some short comings such as unpredictable gravity remove the simulations somewhat from their real world equivalents.
- Path planning - One of the most impacting limitations of this project was the lack of time to execute an effective path planning formula for the robot and drone. While the simple collision avoidance included in Morse was made use of, the robot did not necessarily take the most efficient route to its destination or refuge and collisions for both were not always avoided.
- Robot models - Both the robot and the drone used in these experiments were models that were provided within the Morse package. While they represented close approximations of their real life counterpart do not necessarily represent the same dimensions, weights and in the case of the drone, lifting capabilities.

While these limitations did impact the results of the project, time constraints did not allow for them to be rectified. In all cases there is the opportunity to carry this into future research as discussed below in section 6.

6 Future Work

There are a number of directions that the work begun in this project could be continued. As the subject of robot safety becomes increasingly popular, no doubt further research into a robot's ability to survive predatory encounters will be conducted. Some possible areas of future work are outlined below.

6.1 Further Experiments

This project has proved that flight initiation distance and refuging times have an affect on the success rate and efficiency of a robot in a predator-prey relationship. However, accurately identifying the flight initiation distance and refuge time that would optimise the robot's success rate and efficiency is work that is yet to be done. In order to carry this out, it would be necessary to continue conducting experiments similar to those in this project, with varying behaviours. As such, the ideal flight initiation distance and refuge time for a project of this scope could be identified.

6.2 Varying Factors

One of the downsides in this project is the scope that was covered, which was largely curtailed due to time constraints. This project only looks at one of many scenarios in which robot behaviour in predator-prey relationships affects the robot's success rate and efficiency. The particularities of the experiments carried out in this project only investigate one type of predator and one potential environment. In future research, similar experiments should be carried out varying any of the following:

- environment
- predator
- start and end points
- obstacles

6.3 Mathematical Verification

It was decided that this project should be carried out using simulation, in order to address the hypothesis. However, similar information could have been ascertained using a variety of mathematics methods including pursuit maths. As this method was not selected as the primary method of addressing the hypothesis, it could be used as a verification technique, allowing the results to be tested for consistency.

6.4 Real Life Application

As mentioned above in section 5.3.2 the limitation of using a simulated environment for these experiments is that certain factors may vary from those in real environments. As such, once an ideal flight initiation distance and refuge time are identified, it will be necessary to implement them on a physical robot enduring similar circumstances to ensure no aspects of the simulator affected the results.

6.5 Path Planning

One of the main limitations of these experiments was the lack of time to implement a path planning algorithm, particularly in the case of the robot. Araiza-Illan has carried out investigations into path planning algorithms for robots in similar scenarios and has concluded that the incorporation of sub-goals and a steepest descent search method is the most beneficial [12]. Implementing such a path planning strategy in the robot used in these experiment, would add great value to them and allow for the accumulation of more accurate and realistic results.

6.6 Machine Learning and Familiarity with Predator

One of the most interesting aspects of self-preservation in animals is the way in which animal behaviours develop in conjunction with exposure to the predator. In several studies carried out, it was found that if animals had experience with a species of perceived predator that proved themselves not to be predatory they were subsequently less cautious around said species [18, 14, 15]. Given that this is such a prominent aspect of biological defence in predator-prey relationships, it is something that should certainly be investigated in the world of robotics. Through the use of machine learning, it would be possible for robots to identify potential predators as either threatening or non-threatening based on past experience. This would be particularly useful in situations where a robot is one of a fleet, which is generally the case with delivery robots.

6.7 Predatory Behaviour

A final aspect of biological self-defence strategies that would ideally be incorporated into future work on this topic is an ability to recognise threatening behaviours in potential predators. In these simulations this is done solely based on whether the object interacting with the robot is identified as a drone or not.In biological examples, certain behaviours are considered threatening by prey and will result in greater flight initiation distances, with the prey wanting to maintain a larger distance from the predator at all times [14].This is something that should be take into account when conducting future research in this field.

7 Conclusion

This project has involved a thorough investigation into existing works carried out on self-preservation behaviours in animals and the transfer of these behaviours to robots. This project has carried out a number of experiments designed to investigate the impact that flight initiation distance and refuge time have on the success rate and efficiency of delivery robots in a predator-prey relationship and has reached a conclusion.

The introduction to this paper attempted to give a thorough breakdown of the aims, objectives and scope of this project, so that the intention behind its creation can be fully understood. While it is a complex topic that draws on two vastly different subject areas, the introduction made it clear how the two areas of study will be married effectively. As robots become an increasing presence in our lives, it becomes critical to ensure their own safety. In order to ensure that they are fit to protect themselves, it is necessary to look to biology for the self-preservation techniques used by animals. Attempts to merge this research with robotics proved difficult, but this particular project had definitive enough goals, that making a useful contribution to this end was achievable.

Section three of this paper explored some of the pertinent literature for this project. It began by considering why there is a need for robotic safety and the work that has been done on it so far. This consideration led to the conclusion that emphasis throughout robotic development has been on human safety but, in an era of autonomous vehicles, robot safety needs to come to the forefront. It is imperative that, in the future, robots are capable of protecting themselves, as well as presenting no risk to humans. Following this, an extensive exploration into past self-preservation projects in robots was conducted which indicated that self-aware robots (i.e. robots who were capable of risk-based triggering for self-preservation) tend to have a better chance of avoiding capture when being pursued. It also highlighted the need for a reasonable reaction to predatory behaviour, with the robot being capable of using self-preservation techniques without ignoring its mission. Finally, an investigation into animal escape strategies when being pursued revealed that to have the best chance of escape, an animal must calculate the risk level of a situation and be capable of continuously dynamically assessing the danger they are in throughout the interaction. When calculating flight initiation distance or time spent in refuge, it is necessary for an animal to weigh the cost of staying against the cost of fleeing/emerging. Implementing this in a delivery robot will increase their chances of survival. Based on the information gathered from these explorations, it was concluded that the most important aspects to be tested in this project were the robot's ability to flee at a strategically beneficial time and to remain in its refuge only for the amount of time that it continues to be beneficial. The means by which this has been achieved within the scope of the project have been outlined throughout.

The fourth section of this paper laid out the phases of planning that went into this project. In order for this project to be easily replicated by other researchers it was necessary to highlight the specific implementation requirements that were adhered to in this project. The experiments were split into two sections, depending on the variation of the robot that was being used in them. Each variation of the robot had specific priorities which were implemented in their specific reactions to the predatory encounter. The project worked off the assumption that reacting to a threat is beneficial to a robot's chances of survival based on the findings of [1]. Moreover, this project was conducted on the basis that the ideal self-defence strategy for a ground-based robot being pursued by a cautious, mal-intentioned drone consists of a combination of fleeing and refuging, as was found in [1]. The experiment design of this project was based on a number of factors, which were included to make the simulations as true to life as possible. This included using a realistic environment in which the experiments would take place and ensuring the robot and drone design were as close to life as possible. Simulation was carried out in the Morse simulator and the robot and drone behaviours were programmed in Python. Initially, software design was based on the Morse "Flying Cat and Mouse Game" but this was extensively adapted in order to fit the requirements of this project. Overall, strict project planning resulted in extensive consideration being given to the real life application of these experiments and to the creation of an easily replicable project.

Section five of this paper details the methodology used to carry out the experiments in this project and the results found therein. The methodology of this project was strictly executed and ensured that the experiments were carried out in a controlled manner. Each stage of the simulation was specified which ensured the results of each simulation could be compared as each result was subject to the same factors. The results found over the course of this project make it clear that both the cautious and reckless robots demonstrate the edge cases in terms of flight initiation distance and refuge time. While the cautious robot is almost always successful in completing its mission, it is extremely inefficient. Conversely, the reckless robot has extremely low success rates, however, in cases where it is successful, it is highly efficient. As such, it can be concluded that flight initiation distance and refuge time do indeed impact the success rate and efficiency of a delivery robot in predatory situations. Moreover, neither of the cases examined should be considered ideal flight initiation distance or refuge time as neither combines the optimal success rate with optimal efficiency.

The experiments carried out over the course of this project are relatively robust given the short amount of time allowed for it. However, given more time there are a number of ways that the work in this project could be developed, which are outlined in section six of this paper. Not only could the work carried out in this project be further developed through the execution of a larger number of experiments, additional factors of biological self-preservation could be taken into account in terms of robot safety such as predator behaviour and familiarity with the predator. By implementing these improvements it would be possible to more accurately assess the ideal self-preservation behaviours that should be adopted by delivery robots when avoiding threats.

This project is a necessary addition to bio-robotic development. Based on the study of literature on the subject, it is clear that while projects in a similar area have been conducted, there is no conclusive evidence for ideal behavioural traits in autonomous delivery robots. The experiments carried out in this project were designed to fit into a short time frame and to be easily replicated. The results of the experiments demonstrate that flight initiation distance and refuge time do have an impact on a delivery robot's success rate and efficiency and aid in identifying the ideal behaviours delivery robot's should be equipped with in situations when they encounter a threat.

References

[1] S. K. Chiu, D. Araiza-Illan, and K. Eder. Risk-based triggering of bio-inspired self-preservation to protect robots from threats. *Towards Autonomous Robotic Systems*, pages 166–181, 2017.

[2] M. Giuliani, C. Lenz, T. Mu¨ller, M. Rickert, and A. Knoll. Design principles for safety in human-robot interaction. *International Journal of Social Robotics*, 2010.

[3] S. Dogramadzi, M. E. Giannaccini, C. Harper, R. Sobhani, M.and Woodman, and J. Choung. Environmental hazard analysis - a variant of preliminary hazard analysis for autonomous mobile robots. *Journal of Intelligent Robotic Systems*, 76(1):73–117, 2014.

[4] R. Woodman, A. F. Winfield, C. Harper, and M. Fraser. Building safer robots: Safety driven control. *International Journal of Robotics Research*, 31(13):1603–1626, 2012.

[5] F. Rezazadegan, J. Geng, M. Ghirardi, G. Menga, S. Murè, G. Camuncoli, and M. Demichela. Risked-based design for the physical human-robot interaction (phri): an overview. *Chemical Engineering Transactions*, 43:1249–1254, 2015.

[6] J. Rios-Martinez, A. Spalanzani, and C. Laugier. From proxemics theory to socially-aware navigation: A survey. *International Journal of Social Robotics*, 7(2):137–153, 2015.

[7] D. Vanderelst and A. Winfield. An architecture for ethical robots. 2016.

[8] R.D. Alexander, N.J. Herbert, and T.P. Kelly. The role of the human in an autonomous system. *4th IET International Conference on Systems Safety*, 2009.

[9] P. Salvini and et al. How safe are service robots in urban environments? bullying a robot. *19th International Symposium in Robot and Human Interactive Communication, Viareggio*, pages 1–7, 2010.

[10] A. Castro-Gonza´lez, F. Amirabdollahian, D. Polani, M. Malfaz, and M. A. Salich. Robot self-preservation and adaptation to user preferences in game play, a preliminary study. *Proceedings of the 2011 IEEE International Conference on Robotics and Biomimetics*, 2011.

[11] S. Versteeg and A. Blair. Getting the job done in a hostile environment. *AI '01 Proceedings of the 14th Australian Joint Conference on Artificial Intelligence: Advances in Artificial Intelligence*, pages 507–518.

[12] D. Araiza-Illan. Biologically inspired control systems for autonomous navigation and escape from pursuers. *PhD thesis, University of Sheffield*, 2012.

[13] D.I. Curiac and C. Volosencu. Imparting protean behavior to mobile robots accomplishing patrolling tasks in the presence of adversaries. *Bioinspiration Biomimetics*, 10:1–10, 2015.

[14] T. Stankowich and D.T. Blumstein. Fear in animals: a meta-analysis and review of risk assessment. *Proceedings of the Royal Society of London B: Biological Sciences*, 272:2627–2634, 2005.

[15] T. Stankowich and R.G. Coss. E[U+FB00]ects of predator behavior and proximity on risk assessment by columbian black-tailed deer. *Behavioral Ecology*, 17(2), 2005.

[16] W. E. Cooper, V. Pérez-Mellado, T. A. Baird, J. P. Caldwell, and L. J. Vitt. Effects of risk, cost, and their interaction on optimal escape by non-refuging bonaire whiptail lizards. *Cnemidophorus murinus, Behavioral Ecology*, 14(2):288–293, 2003.

[17] J. Martin and P. Lopez. When to come out from a refuge: risk-sensitive and state-dependent decisions in an alpine lizard. *Behavioural Ecology*, 10(5):487–492, 1999.

[18] E. Reimers, L. E. Loe, S. Eftestøl, J. E. Colman, and B Dahle. Effects of hunting on response behaviors of wild reindeer. *Journal of Wildlife Management*, 73(6):844–851, 2009.

A Appendix - Reckless robot results

Battery Time Capture Hid Delay
97.048 42.91 Y Y 26
97.872 31.1 Y Y 12
97.839 31.22 Y Y 12
95.499 39.23 Y Y 21
96.572 35.84 Y Y 17
96.09 62.93 N Y 1
97.27 44 Y Y 28
97.55 26.67 Y Y 3
96.962 48.13 Y Y 30
97.357 32.08 Y Y 13
97.338 42.34 Y Y 26
97.458 39.99 Y Y 22
98.1 27.79 Y Y 7
96.447 44.45 Y Y 28
96.942 47.66 Y Y 30
96.521 53.26 Y Y - REACHED 4
97.274 43.87 Y Y 27
96.569 46.64 Y Y 30
95.888 42.37 Y Y 26
97.817 32.97 Y Y 18
97.36715987 42.19070387 Y Y 26
97.59901384 37.75519133 Y Y 21
97.38765045 41.75170636 Y Y 25
97.18453876 45.92805791 Y Y 29
97.7057815 35.47188783 Y Y 18
97.68861737 35.76703382 Y Y 19
97.40760167 41.45039487 Y Y 25
97.47711196 40.03395343 Y Y 23
97.70603019 35.39907384 Y Y 18
97.64378914 36.69454765 Y Y 23
97.19079694 45.97364974 Y Y 30
97.18218829 45.90206599 Y Y 29
97.32044312 43.18566871 Y Y 27
97.74494059 34.70908237 Y Y 17
98.29706688 23.59900069 Y Y 2
98.14888438 26.55690479 Y Y 7
97.36722654 42.16902685 Y Y 26

96.37855529 62.07470274 Y Y - REACHED 19
97.84148654 32.69404078 Y Y 14
95.54796556 78.69668603 N Y - REACHED 19
97.4862734 40.82932901 Y Y 24
96.43319318 60.98168135 N Y - REACHED 19
97.87862785 31.99933004 Y Y 12
97.48918994 39.8279705 Y Y 23
98.16125261 26.51555371 Y Y 7
98.69856006 15.58232403 Y Y 1
98.10962315 27.44830585 Y Y 8
97.36268277 42.23335028 Y Y 26
98.68886285 15.72669673 Y Y 1
97.54855215 38.57383966 Y Y 19
96.70010942 55.52821493 Y Y - REACHED 5
97.54943505 38.54561353 Y Y 22
97.97296695 30.04877281 Y Y 11
97.59144778 37.78993416 Y Y 21
97.70533513 35.42216039 Y Y 18
97.18183955 45.93938732 Y Y 29
97.19446413 45.66574454 Y Y 29
95.54902186 78.68304992 N Y 19
97.58871174 37.69745779 Y Y 21
97.54582024 38.70521617 Y Y 22
97.94664506 30.9925015 Y Y 12
95.41456708 81.23607278 Y Y - REACHED 2
97.08996883 47.74708223 Y Y 30
98.3079781 23.36649919 Y Y 4
97.26065266 44.27632856 Y Y 28
97.08777982 47.76817703 Y Y 30
97.48322114 39.78170133 Y Y 23
97.26304195 44.28645849 Y Y 28
96.87217705 52.09068656 Y Y - REACHED 5
97.89364781 31.77879047 Y Y 13
97.97217586 30.08004212 Y Y 11
97.31830226 43.18820786 Y Y 27
97.38785705 41.72812796 Y Y 26
96.80717341 53.34726381 Y Y - REACHED 3
97.75080896 34.53775382 Y Y 18
98.14461056 26.54766274 Y Y 7
97.39257032 41.76951814 Y Y 25

97.69461594 35.59342337 Y Y 18
98.10206689 27.46814561 Y Y 8
97.75057908 34.56102228 Y Y 16
96.81306834 53.19366455 Y Y - REACHED 5
97.74019312 34.74209046 Y Y 16
97.84215429 32.67535234 Y Y 14
98.69379365 15.59572721 Y Y 1
97.38845038 41.72650909 Y Y 25
97.84137084 32.6996839 Y Y 14
97.3885859 41.686795 Y Y 25
97.84165302 32.70669127 Y Y 14
97.57816011 37.9329927 Y Y 21
97.26904762 44.18944693 Y Y 28
97.74677747 34.56684542 Y Y 16
97.17826889 45.95857835 Y Y 29
97.97376479 30.06622672 Y Y 11
97.18598073 45.89269733 Y Y 29
97.18109086 45.79109502 Y Y 29
96.84260083 52.75094652 Y Y - REACHED 5
98.19710682 25.62941003 Y Y 6
97.83806846 32.71182823 Y Y 14
98.01939824 29.12573624 Y Y 10
97.13573917 46.80895567 Y Y 30