

Risk Informed UAV Pathfinding, UPX

Micko Lesmana

11201901010



Undergraduated Thesis

Department of Aviation Engineering
International University Liaison Indonesia
BSD City
3 December 2022

Approval Page

Report

Risk Informed UAV Pathfinding, UPX

3 December 2022

Micko Lesmana

11201901010

Department of Aviation Engineering

Acknowledge by:

CONTENTS

1	Introduction	3
1.1	Introduction	3
1.2	Research purpose	3
1.3	Research scope	3
1.4	Research problem	4
1.5	Significance of the study	4
1.6	Research question	4
2	Literature Review	5
2.1	Ground risk model in UAV operation	5
2.2	Components of ground risk model	5
2.2.1	Failure model	6
2.2.2	Impact location model	7
2.2.3	Recovery model	8
2.2.4	Stress model	8
2.2.5	Exposure model	9
2.2.6	Incident stress model	9
2.2.7	Harm model	10
2.3	Risk informed path finder	11
3	Background Theory and Methodology	13
3.1	General Overview	13
3.2	Prerequisite data	13
3.2.1	Online geographic data	13
3.2.2	Population data	14

3.2.3	Geographic Information System	14
3.3	Risk map	15
3.3.1	Georeference layers	15
3.3.2	Population density layer	15
3.3.3	Obstacle layer	16
3.3.4	Sheltering factor layer	17
3.3.5	No-fly zone layer	18
3.4	Ground Risk Assessment	19
3.4.1	Failure rate	20
3.4.2	Descent Event	20
3.4.3	Ballistic Descent	21
3.4.4	Glide Descent	21
3.4.5	Fly away	22
3.4.6	Parachute	23
3.4.7	Wind effect	24
3.4.8	Probability of Impacting an Individual	24
3.4.9	Probability of Fatality	25
3.5	Finding optimal path	26
3.5.1	Problem statement	26
3.5.2	Dijkstra's algorithm	28
3.5.3	A* algorithm	29
3.6	UPX, UAV Pathfinding eXtension	33
3.6.1	Limitation	34
3.7	UPX Python and packages	34
3.7.1	Python	34
3.7.2	GeoPandas	34
3.7.3	Shapely	35

3.7.4	Tesspy	35
3.7.5	OSMnx	35
3.7.6	NetworkX	35
4	Results	36
4.1	Ground risk map generation	36
4.1.1	UAVs Specification	36
4.2	Objective	37
4.3	PDF risk map	38
4.3.1	Talon	38
4.3.2	Parrot Disco	41
4.3.3	DJI Phantom 4	43
4.3.4	DJI Inspire 2	46
4.4	Pathfinder result	48
4.4.1	Regular A*	48
4.4.2	Risk A*	49
4.4.3	Risk result	50
5	Summary, Conclusion, and Recommendation	54
5.1	Summary	54
5.2	Conclusion	54
5.3	Recommendation	55
5.3.1	For future research	55
5.3.2	For software design	55

CHAPTER 1

INTRODUCTION

1.1 Introduction

Unmanned Aerial vehicles (UAVs) also known as a drone, is getting more recognition in recent years. The booming of UAVs is caused by the component needed to manufacture the UAV, especially microcontrollers getting cheaper, and smaller. Compare to previous UAV iteration, UAV nowadays offers much wider capability, such as enhanced range, the ability to mount hardware, and camera for example, and of course easily deployable without requiring many technicians to maintain them.

UAVs are often being used in area where it is dangerous, impractical, and expensive for human. The affordable entry point of using a UAV incentivized emerging industries to exploit the capability that UAVs offer. Not only that but the existing industry is also getting interested to use UAVs as an augmentation to its existing operational infrastructure. And that industries are, logistics companies, agricultural companies, multimedia companies, and surveyors to name a few

Although it provides a great solution, UAVs have the potential to pose risks to people, property, and other aircraft. Many countries have regulations in place that require UAVs to be operated in a safe and responsible manner. Risk-based UAVs path planner can help to ensure that UAVs are operated safely and responsibly. By considering potential risks and hazards, a risk-based approach can help to reduce the likelihood of accidents or incidents occurring during the flight. By considering potential risks and hazards, this approach can help to reduce the likelihood of accidents or incidents occurring during the

Other than its impacts towards public, UAVs can also potentially impact the environment, for example by disturbing wildlife or damaging natural habitats. UAV path planner can help to minimize these impacts by establishing rules around the use of UAVs in sensitive areas.

1.2 Research purpose

For ensuring the safety and efficiency of UAV operations, this research aims to develop a UAV trajectory planning based on the risk assessment on the ground.

1.3 Research scope

- UAVs model that being used in this research are fixed wing and multirotor.
- This research only use point mass representation with generalize aerodynamic equation.
- Based on the point above, UAVs failure is assumed to be only power failure,

without damage to its shape.

- Risk assessed in this paper are combination of human exposed area and area that can cause major accident such as airfields, and power station.
- This research use Jakarta, Indonesia as its geographic location, since it has varying building density, rivers, and random patch of lands.
- A* and Djikstra pathfinding algorithm are being used for this research.

1.4 Research problem

- There are risks associated when of UAV falls on the ground.
- Modeling physical interaction of UAV crash.
- Selection of pathfinding algorithm to get optimum time.

1.5 Significance of the study

UAV Pathfinder application could be one of the toolkit for UAV operator to plan their safe operation of flying their UAVs, reducing incident and accident. Developer and researcher could use the library itself as an extension or starting point in the development of the UAS Traffic Management application.

1.6 Research question

- Which kind of geographic data and data processing model perfectly suited for pathfinding risk assesment.
- How to statistically create the best approximation of crash dynamics.
- What kind of pathfinding method that produces the quasi-optimal path without sacrificing computational time.

CHAPTER 2

LITERATURE REVIEW

2.1 Ground risk model in UAV operation

A ground risk model is a type of assessment tool that is used to evaluate the potential risks associated with the operation of UAVs. This model takes into account a variety of factors, including the physical environment in which the UAV will be operating, the capabilities, and the limitations of the UAV itself. The goal of a ground risk model is to identify and prioritize potential risks in order to produce informed decision-making and risk management strategies for UAV operations.

There are a number of different approaches that have been taken to develop the ground risk models for UAVs. One approach is the use of probabilistic risk assessment methods, which involve the quantification of risks using probability and consequence metrics. Washington et al., 2017 compiled many methods to developed subcomponent of risk analysis in georeference approach. Another approach is the use of scenario-based methods, which involve the development of detailed scenarios that describe potential UAV operations and the associated risks. The scenario usually coupled with safety and hazard level per operation basis. This method is useful for preliminary assessment using method such as risk tree diagram, bayesian belief network Ancel et al., 2017, and hazard risk assessment matrix Barr et al., 2017. Although one might seem suggested both method produce difference analysis but from overall point of view they describe the same phenomena.

Overall, the use of ground risk models has the potential to significantly improve the safety and effectiveness of UAV operations. By identifying and prioritizing potential risks, ground risk models can help operators make informed decisions about how to mitigate or manage those risks. As the use of UAVs continues to grow and expand into new applications, the importance of ground risk modeling is likely to increase as well.

2.2 Components of ground risk model

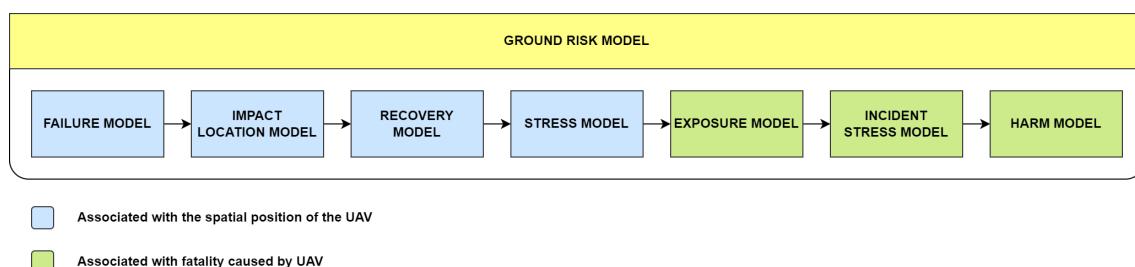


Figure 2.1: Components of ground risk model

A ground risk model for unmanned aerial vehicles (UAVs) typically consists of several sub-components that are used to assess the various factors that can impact the safety and effectiveness of UAV operations. However, this approach is not new, since many earlier researches already follow a guideline to model these components. Breunig et al., n.d. provide general guideline for the model, while Washington et al., 2017 provide a detailed description for the model. Washington et al., 2017 method of compilation is based on Pat-Cornell, n.d., which studied the level of analytical detail when modeling the risk formulation. The components in ground risk model are, failure model, impact location model, recovery model, stress model, exposure model, incident stress model, and harm model.

$$P_{fatal}(x) = P_{event}(x) \cdot P_{impact}(x) \cdot P_{incident}(x) \cdot P_{harm}(x) \quad (2.1)$$

All of those models are represented as a result of probability of said event-model occurring. Since all models are consecutively occurring from failure model to harm model, each component of the ground risk model are being multiplied by one and another. Although in many papers, not all researcher use all of these components in their ground risk model, and often time use different methodology to analyze the same component. For example, Primatesta et al., 2020 only use event, impact location, and fatality model, while Kim and Bae, 2022 use event, impact, exposure, incident stress, and harm model. Despite in a glance that there are difference in modeling the ground risk, sometimes the researcher combine multiple subcomponent as one. Again for example in Primatesta et al., 2020 fatality model component is in fact a combination of incident stress and

2.2.1 Failure model

A failure model is the first subcomponent in the ground risk model that evaluates the potential for failures in UAVs. This potentially leads to a crash impact in a given variety of factors that can contribute to UAV failures, including design and manufacturing defects, operational errors, and environmental factors.

The failure model can help to understand on how the UAVs would behave after experiencing the failure, Washington et al., 2017 described four distinct events following the failure. Which are Unpremediated descent scenario (UDS), Lost of control (LOC), controlled flight into terrain (CFIT), dropped or jettisoned component(DOJC).

$$MTBF = \frac{\rho\pi b^2}{2.78} = k(\rho\pi b^2) \quad (2.2)$$

Likelihoodness of failure rate described on failure model is divided into two type of formulation. The first one is based on historical data and the other one is based on the expert opinion. Data driven historical data are using previous UAVs crash, and its UAVs specification as a starting point for data interpolation and extrapolation for calculating mean time between failure (MTBF) Breunig et al., n.d. and Weibel and Hansman, n.d. for example. However, since UAV industries is in its infancy and an added fact with many variants of UAV already exist in short period of time, resulting in data reliability issues. Therefore, it may be more appropriate for UAS to utilize a functional approach to calculate

the system failure rate. Examples of functional and structural decompositions of UAS that may be used to create such a model are given Burke et al., 2011.

To assess the system's dependability, other methods can be utilized, such as Fault Tree Analysis (FTA), Event Tree Analysis (ETA), and Failure Mode, Effects and Criticality Analysis (FMECA) Barr et al., 2017. Bayesian Belief Networks (BBN) Ancel et al., 2017 have also been used more recently in this regard. Ancel et al., 2017 provide the most detailed failure risk model incorporating real-time monitoring of UAV component status to determine the likelihoodness of failure.

All the models also use the assumption that the failure's probability of happening is constant. Therefore, variables like phase, length, environment circumstances, or mission profile that might affect the chance of failure are not taken into account.

2.2.2 Impact location model

The impact location model describes the position and size of the impact region caused by a UAV crash. Some models propose single-point impact, while others consider that, given a specific trajectory, the UAV may make contact at a number of different locations. The latter strategy establishes a perimeter around which all of the UAV's possible impact points are located. The perimeter of this distribution is calculated based on the possible maximum UAV descent trajectory distance.

The impact location model can be formulated using two different models, hypothetical Ball et al., n.d., Burke et al., 2011, Primatesta et al., 2020, Primatesta et al., 2020 and empirical model.

The hypothetical model uses the aerodynamic model to simulate the behavior of the impact location. This model can be further separated into a ballistic descent, gliding descent, and planform model. The gliding model such as in assumes the UAVs will continue descending based on their glide profile. The planform model uses the aircraft wings and its layout to calculate the descent trajectory. And the ballistic descent calculated the trajectory based on the ballistic profile. Most research use gliding and ballistic descent to calculate the impact model.

$$A_{exp}(\theta) = \pi(r_p + r_{uav})^2 \sin(\theta) + (r_p + r_{uav})(h_p + r_{uav})\cos(\theta) \quad (2.3)$$

Primatesta et al., 2020 added parachute descent, an event where UAVs deployed its parachute when descending. And fly away descent, where the pilot lost control of the UAV, while the UAV automatically maintains a stable cruise.

Wu and Clothier, 2012 use a dynamic model assessed for various failure conditions to determine the maximum boundary (and in turn area) of the impact distribution for fixed-wing UAVs. However, like most models, it assumes a uniform impact distribution within these boundaries.

While an empirical model using historical aircraft events is used as a starting point

by interpolation and extrapolation algorithms to pinpoint the impact site. Melnyk et al., 2014 further classifies these models into weight based on the study conducted by Ale and Piers, 2000

2.2.3 Recovery model

A recovery risk model for UAVs is a model used to assess the likelihood of successful recovery of a UAV to a optimal or degraded operational state following a failure or malfunction. The model takes into account a variety of factors, the nature and severity of the failure, and the environment in which the recovery is taking place. The recovery risk model also takes into account the UAV design, capabilities and limitations. For example, a fixed wing UAV will have different recovery options as compared to a multirotor UAV. Also, the flight control system, navigation and sensor capabilities play an important role in the recovery process.

The type of failure, its method of failure, and the operating environment all influence the likelihood of recovery. For instance, if the UAVs malfunctions and enters a lost-of-control failure state, the existence of a recovery mechanism in this case, a parachute, would slow down the drop velocity and lessen any potentially damaging effects of the malfunction. Unfortunately many ground risk models omitted the use of the recovery model, with small exception such as the model provided by Shelley, 2016, Bleier et al., 2015 and Primatesta et al., 2020 explore the effectiveness of parachutes. As concluded, recovery parachute descent employed by Primatesta et al., 2020 shows significant decrease in the kinetic impact energy, and reduce the overall fatality.

2.2.4 Stress model

The stress model explains the unpredictability of harmful circumstances occurring at a specific time and place. Depending on the kind of harm being characterized as part of the risk assessment, a specific harmful situation is modelled. For instance, stresses may be described as kinetic energy if the outcome to be calculated was human physical harm (KE). While secondary effects such as explosions and the release of hazardous chemicals will have distinct related stress qualities that need to be modeled differently. There are model proposed by Ball et al., n.d. to analyze the consequences of heat radiation and explosions which often produces by chemical reaction.

Depending on the kind of UAV, different stress factors that relate to one or more harmful processes may exist. Additionally, the UAV's characteristics and its trajectory may be changed to alter the stress profile at the site of impact. Through one or more harm mechanisms, the various stresses may be linked to the undesirable consequence. Examples of physical injury to humans include blunt force, penetration, crushing, blast, burns, lacerations.

It should be mentioned that trauma generated by a blunt force collision is the main cause of injury taken into account by current models. It is possible that blunt force trauma is not the main cause of injury for UAV with tiny mass. Smaller multirotor UAV, for in-

stance, have a higher tendency to lacerate individuals and inflict physical harm. In the stress model, laceration injuries must be taken into account, according to the Small, 2009.

The KE at impact was calculated by the models using a variety of mass and speed values, which is why the results of the stress models can differ from one another. For instance, Ancel et al., 2017 and other models simply assume the maximum velocity for estimating the impact energy for small UAV flying at low altitudes. Although using terminal velocity would be an overly conservative estimate, Dalamagkidis et al., 2008 evaluation of the impact KE employs a similar assumption.

2.2.5 Exposure model

An exposure model is a method for determining how affected individual or property may be to dangers related to the use of UAVs.

A uniform exposure model is the model of population exposure that is most frequently utilized. The use of this approach overlooks possible population clustering within a specified geographic region and, as a result, averages the potential exposure factors contribution to the overall risk assessment. Such methodologies tend to overestimate or underestimate the risk value associated with the current setup because they do not provide sufficient information about the spatial and temporal distribution of the population.

Models such as those created by Burke et al., 2011 and Melnyk et al., 2014 aim to add temporal features, acknowledging how individuals spend their days, taking such factors into consideration. An illustration of a thorough exposure model that emphasizes the time dependencies in population exposure is given by Melnyk et al., 2014. Research conducted by Klepeis et al., 2001 found that individuals spent 68.7% of their time at home, 7.6% outside, 5.5% in cars, 5.4% at factories or offices, and another 12.8% in various indoor settings. This data is used by Melnyk et al., 2014 to specify their exposure model along with the information.

As expected population density data gathered for exposure model came from census, geographical data from satellite imagery. Depending on when each study was done and what regions were being analyzed, a different census data source was used. For instance, Burke et al., 2011 used data from the U.S. Census Bureau, and Clothier et al., 2007 used data from the Australian Bureau of Statistics. And to provide non uniform population data, cellular networks were used by Ancel et al., 2017 to create a near-real-time model of the population distribution.

2.2.6 Incident stress model

The incident stress model describes how much stress an individual is exposed to. However, a large number of the discovered ground risk models do not take event stress variance into consideration. Therefore, it is presumed that the UAVs KE is entirely directed directly at the human. Although a conservative assumption, it may cause the degree of risk to be overestimated. Not all studies adopt the sheltering model since most information about the material composition of shelters is generally limited to local government

databases. limiting the range of potential general approaches to building shelter-based models.

On the other hand, the model that takes into account an attenuating factor uses a straightforward KE model based on the sheltering effect to calculate how much kinetic energy a certain building can hold. An existing study of debris was used to develop the sheltering model.

In the model described by Ball et al., n.d., the amount of KE that is absorbed by the shelter as a result of its deformation, breaking, and movement as a result of an incident KE is subtracted from the overall risk posed by the system.

Based on investigations undertaken by the Columbia Accident and Investigation Board and the Department of Defense, Melnyk et al., 2014 describe a method for calculating the risk from population distribution within various types of shelters, mostly residential structures and commercial buildings.

2.2.7 Harm model

The harm model describe the severity of UAV impact towards the population. The harm model usually measured in probability of fatality given UAV impact to a person. The impact harm model can be categorized based on its severity from minor, major, and fatal injury. Many literature paper use impact energy as a variable when it comes to determining the fatality imposed by the UAV. This impact energy can be attenuate depends on the protective shelter arround the population(e.g buildings, trees, and vehicles). The mechanic of injury also varied from blunt force, penetration, laceration, and chemical burns.

Of every injury mechanism, the most common studied harm mechanism is blunt force trauma. According to Shelley, 2016, the most likely impact injuries are injuries to the head, particularly skull fracture.

$$f(x) = \frac{1}{1 + e^{-\frac{(x - threshold)}{energy\ impact}}} \quad (2.4)$$

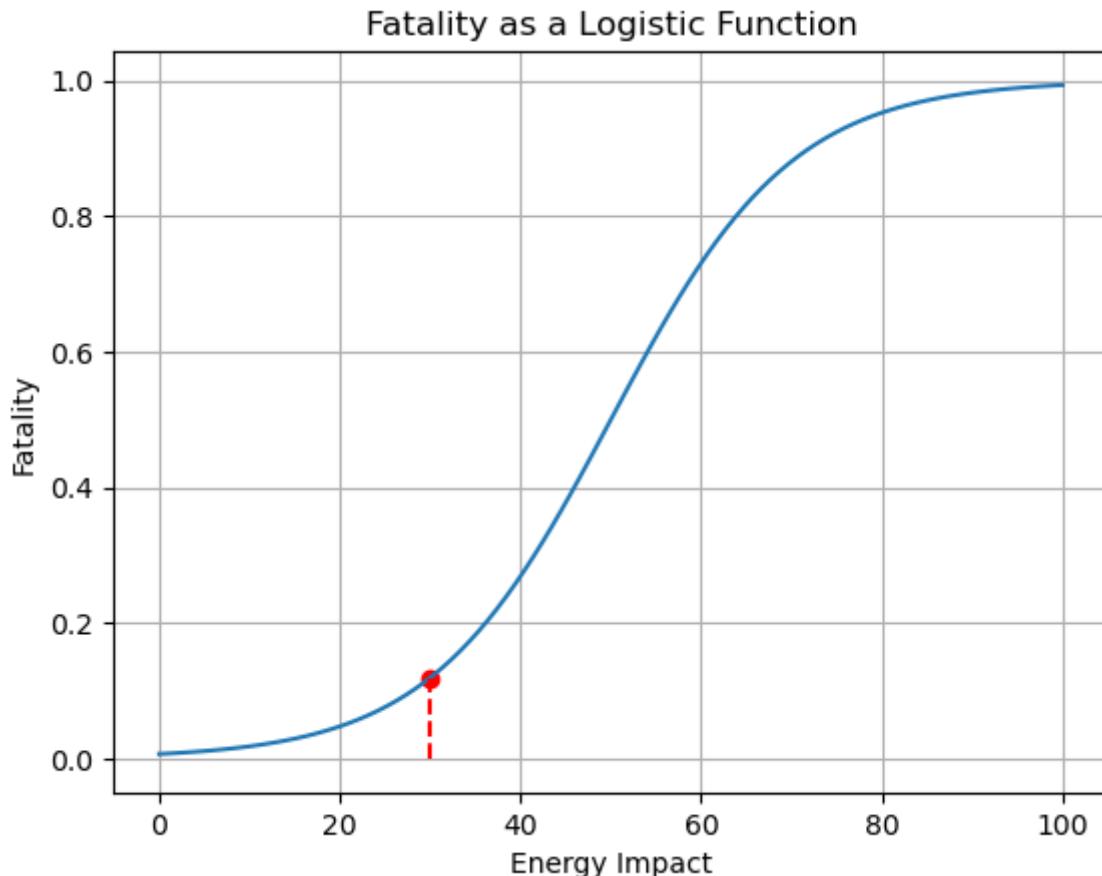


Figure 2.2: Generalized logistic fatality function given energy impact

For various impact locations, the characteristic of fatality probability model have been developed as a function of incident KE Harwick et al., 2007. The model describes the harm response of an average male, averaged over varying impact orientations, as a standard logistic curve probability of fatality described in Ref. Shelley, 2016 is presented in Eqn. 2.4. Where threshold is the impact energy associated with a 50% probability of a fatality (measured in Joules), Energy impact is the impact energy. A number of advancements over this standard equation have been presented by Dalamagkidis et al., 2008 and Shelley, 2016 to name a few.

Magister, 2010 makes use of a blunt criteria (BC) that relates the kinetic energy on impact with the body's ability to tolerate the energy on impact, which is expressed using the Abbreviated Injury Scale (AIS).

2.3 Risk informed path finder

This study not only for generating the ground risk map for the risk assessment but also measures and finds the path taken by the aircraft to find the least amount of risk and distance traveled required.

There are multiple pathfinders, or in general, a method that is being used in the UAV industry as a navigation algorithm. All of these algorithms have different criteria, operating principles, and goals to complete.

Navigation algorithms are generally categorized into 2 parts based on the operational time: pre-operational algorithm and real-time operational algorithm.

The pre-operational algorithm defines a general pathway that is going to be inserted into the UAV flight controller. At the same time, the real-time operational algorithm is the algorithm where the UAV actively calculates the best direction based on its predefined pathway obtained through a pre-operational algorithm and real-time sensor data to avoid danger and risk.

RRT, or rapidly exploring random tree, is a navigation algorithm for motion planning. It is usually employed as an approximate method for robots to traverse the real world, which is full of dynamic change in the environment. This RRT algorithm works by expanding the space tree, where the space, in this case, is a node. If the UAV needs to go from point A to D, the UAV then tries to expand the search tree to find the feasible series of nodes to traverse through. This RRT can be inserted into various constraints, in this case, risk, where the UAV tries to find the branch with the least amount of risk.

A* algorithm is one such algorithm that works under the principle of the pre-operational algorithm. A* is a pathfinder algorithm to find the least amount of distance to move into the final position.

A* itself is an extension of Djikstra's algorithm, where both of them use the so-called cost function $f(x)$, where the cost function is usually a distance cost. We can extend this concept by multiplying by the risk coefficient to create a modified Risk A* algorithm, which it calculates the optimal distance and risk involved in the operation.

A study conducted by Primatesta et al., 2020 and Clothier et al., 2007 shows how to implement the risk A* algorithm when developing ground risk management. However, a study conducted by Primatesta gives more detail in implementing the algorithm itself, while Wu and Clothier implement a much more simpler method by separating and calculating independent A* for risk and distance.

Since the ground risk model for this study is pre-operational, the Risk A* method would be chosen to calculate the general distance and risk score of the pathway.

CHAPTER 3

BACKGROUND THEORY AND METHODOLOGY

3.1 General Overview

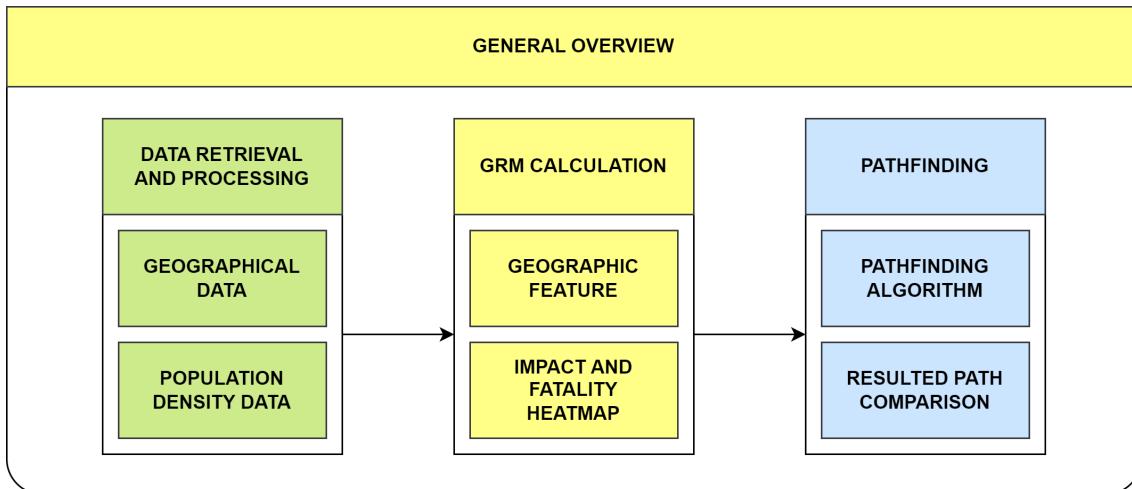


Figure 3.1: General overview

In general, there are three main objectives to create an optimal risk-aware flight path. These are data retrieval and processing, ground risk map calculation, and generating the most optimal path using the pathfinder algorithm.

To collect necessary data, we are using free services provided by OSM(OpenStreetMap)“Open-StreetMap Wiki”, n.d. to retrieve geographic data, and GHSL(Global Human Settlement Layer)“Global Human Settlement - Mission - European Commission”, n.d. to retrieve population density throughout the earth.

The GRM is calculated using geographic and population data, and also the type of aircraft that are being used in the calculation. The result would be descent location, impact energy, and fatality probability heat map.

After we compute and discretized the GRM. We find the optimal path with the least amount of risk using pathfinding algorithms, where the result would be compared.

3.2 Prerequisite data

3.2.1 Online geographic data

OSM “OpenStreetMap Wiki”, n.d. is a collaborative project that aims to create a free, editable map of the world. It was founded in 2004 and has since grown to become one of the largest and most popular sources of geospatial data.

OSM is built on the idea that anyone can contribute to the map by adding and updating data. Users can add data using GPS devices, aerial imagery, or by manually tracing over satellite imagery. The data is then stored in a database and made available under an open license, allowing anyone to use and share the data.

There are multiple ways to use the OSM service. To retrieve the geographic data we can send the query to Overpass API and receive the default geoJSON data. OSM also provide geocoding and reverse geocoding using Nominatim API.

3.2.2 Population data

GHSL “Global Human Settlement - Mission - European Commission”, n.d. It is a dataset produced by the JRC (Joint Research Centre) of the European Commission that offers details on the distribution and location of human settlements worldwide.

The GHSL classifies and examines patterns of urbanization and human settlement using new spatial data mining technologies and remote sensing data, such as satellite images. The dataset is periodically updated to reflect changes in urbanization and population growth and contains data on built-up areas, population density, and other indices of human presence.

Overall, the GHSL is a valuable tool for comprehending the spatial dynamics and patterns of human settlement and can assist planners in making better-educated choices concerning urbanization and sustainable development.

3.2.3 Geographic Information System

Geographic and population data are then stored and processed by GIS(Geographic Information Systems), tools used for storing, manipulating, and visualizing geographic data. They allow users to analyze and understand spatial relationships, patterns, and trends in geographic data.

GIS consists of several key components:

- Data: GIS uses data that is linked to a geographic location. This data can be in the form of points, lines, or polygons (areas) and can include attributes such as population, land use, or elevation.
- Hardware and software: GIS requires specialized hardware and software to store and analyze the data. This includes computers, servers, and specialized software such as ArcGIS or QGIS.
- Maps: GIS allows users to create interactive maps that can be used to visualize and analyze the data. Maps can be customized with different layers of information, such as roads, land use, and population density.
- Analysis: GIS allows users to perform spatial analysis on the data, such as calculating the distance between two locations or identifying patterns in the data.

There are multiple GIS software suites, from standalone programs such as ArcGIS, and QGIS, to package base library for python, such as GeoPandas, Rasterio. In this paper we are going to use GeoPandas and Rasterio and its supporting libraries.

3.3 Risk map

The Risk map is a graphic representation depicting a potential risk associated with the UAV operation. Risk maps are divided equally in distance into cells. Each cell represented a bounded geographical location. The map can be represented by a $M \times N$ size matrix, which each $M \times N$ is the number of cells in each row and column respectively. Each $\mathbf{R}(i, j)$ cell has centroid coordinates which represented the geographic location of (x, y) in any coordinate reference system (CRS).

If we use a slight notation abuse, which is $\mathbf{R}(i, j)$, each cell coordinate (i, j) is represented as a geographical location in which case shifted by x and y amount. Figure [Dex] illustrates the Risk map matrix.

3.3.1 Georeference layers

In a sense risk map is constructed by stacking layer of features, each layer itself is obtain by analyzing each geographical feature and population density to calculate the ground risk map. In this paper, multilayer frameworks are used based on the work of Primatesta et al., 2020. As the name implied, this framework contains multiple georeference layers. These layers are :

- **Population density layer:** Describes population density in the area.
- **Obstacle layer:** Describes geographic features that can block aircraft flight paths such as building height, or any other obstacle.
- **Sheltering factor layer:** Describes geographic features that provide shelter for humans in the area.
- **No-Fly zone layer:** Describes geographic features in which aircraft cannot enter, such as military base, or airport

3.3.2 Population density layer

The population density layer describes the population density and distribution in the area. It is one of the most crucial factors in the risk assessment is population density since it indicates how many individuals can be affected by a vehicle crash and how they are dispersed over the map.

To conduct an accurate risk assessment it is necessary to provide population density data with fine resolution. since the likelihood of colliding with a person on the ground is particularly impacted by population density.

The population data are frequently gathered by the government human settlement institute. However, this approach is not easily ported from country to country since different institutes produce population data differently. As previously mentioned above GHSL would be used in this paper, since it already has the global database.

In this study, the population density layer is represented by a 2D georeference map with each cell corresponding to the value of population per cell area size with cell area in unit m^2 , therefore people/m^2 . Logically, the map would be in matrix form, \mathbf{D} , with each cell element of $\mathbf{D}(x, y)$.

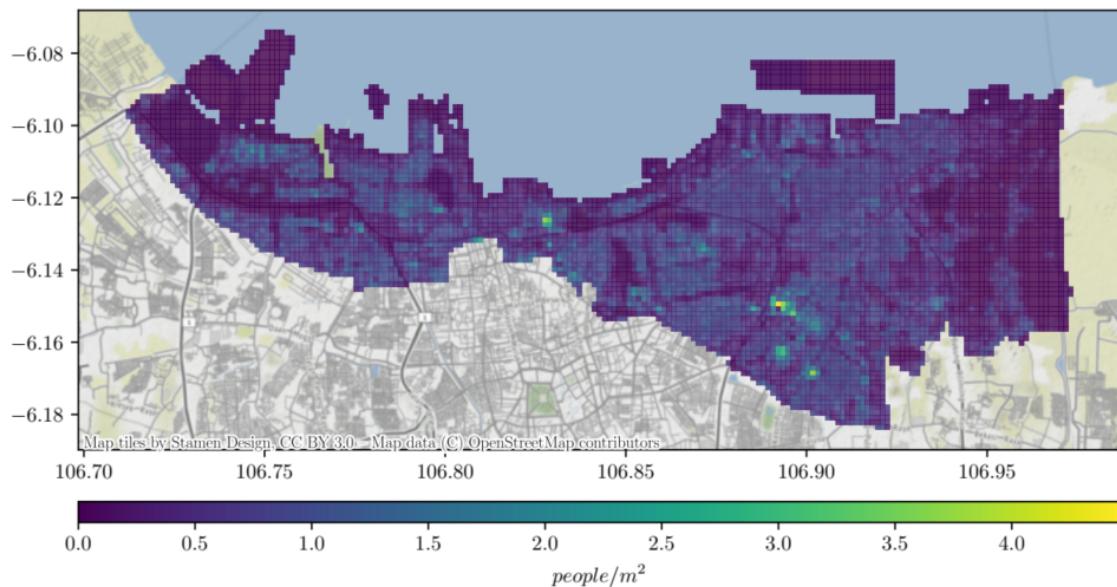


Figure 3.2: Population density map

3.3.3 Obstacle layer

Obstacle layer describe which cell has contains a building or geographic feature which higher than flight altitude of UAV. the data can easily be gathered by filtering, and mark it as impasable terrain. since it is just another georeference layer, we can represent the as a matrix $\mathbf{O}(x, y)$ which describe maximum height of building in a given cells.

Noted that this layer does not explicitly accounted into risk factor analysis, but instead only for determining each cell is impasable for path finder.

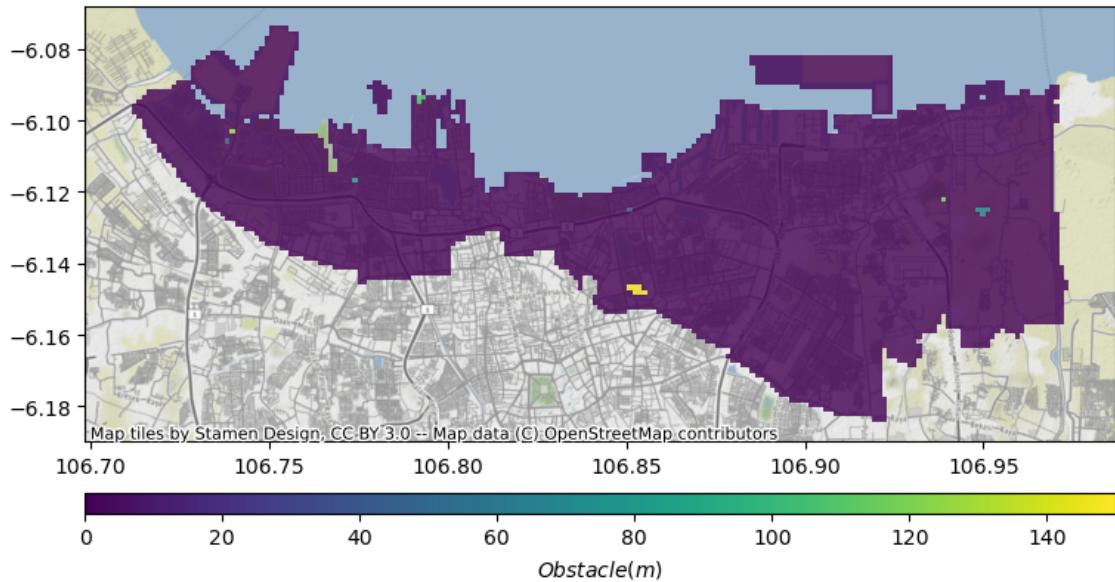


Figure 3.3: Height obstacle map

3.3.4 Sheltering factor layer

Sheltering Factor layer is a term for describing, the layer that focuses on assessing the presence and characteristics of physical structures or objects that can provide shelter. It is represented with a matrix \mathbf{S} , where each element $\mathbf{S}(x, y)$ assumes the sheltering factor value of the corresponding location.

The sheltering factor layer takes into account factors such as the size, shape, material, and location of the structures or objects in the drone's operating environment. It aims to assess how these factors can influence the risk of fatality in case of failure.

Unfortunately, it is impractical to evaluate the sheltering element in full detail in a study of this kind because it is a very complex scenario. The dynamic of drone collision is a complication in itself, particularly since detailed physical building features, like material or wall thickness, are recorded differently around the world.

It is worsened by the fact that there are multiple studies with conflicting results when calculating the absorption of shelter since there is no consensus when defining the sheltering factor. Some studies by Dalamagkidis et al., 2008 and Primatesta et al., 2020 use a scale factor from 0 to 1, or 1 to 10 respectively, representing the energy absorption factor of a given building. While Melnyk et al., 2014 uses a study conducted by the US Department of Defense (DoD) Harwick et al., 2007 regarding explosive study to measure the energy absorption capability.

Even though it is tempting to use the method given by Melnyk et al., 2014 and DoD since it has the most accurate data. The study conducted by them is more suitable for singular operation-specific evaluation i.e. best case use for generating event tree, and not suited for generating risk map. There are also multiple problems when using this method, such as the lack of accurate data on building material geographic maps, espe-

cially when OSM do not provide a building material data, which would result in inaccurate measurement.

From the previous statement, in this paper, we would use the scale factor proposed by Klepeis et al., 2001 and Primatesta et al., 2020 as a foundation for sheltering factor risk generation. Since it provides a general method to calculate a risk factor. Table 3.1 is the shelter factor for types of features.

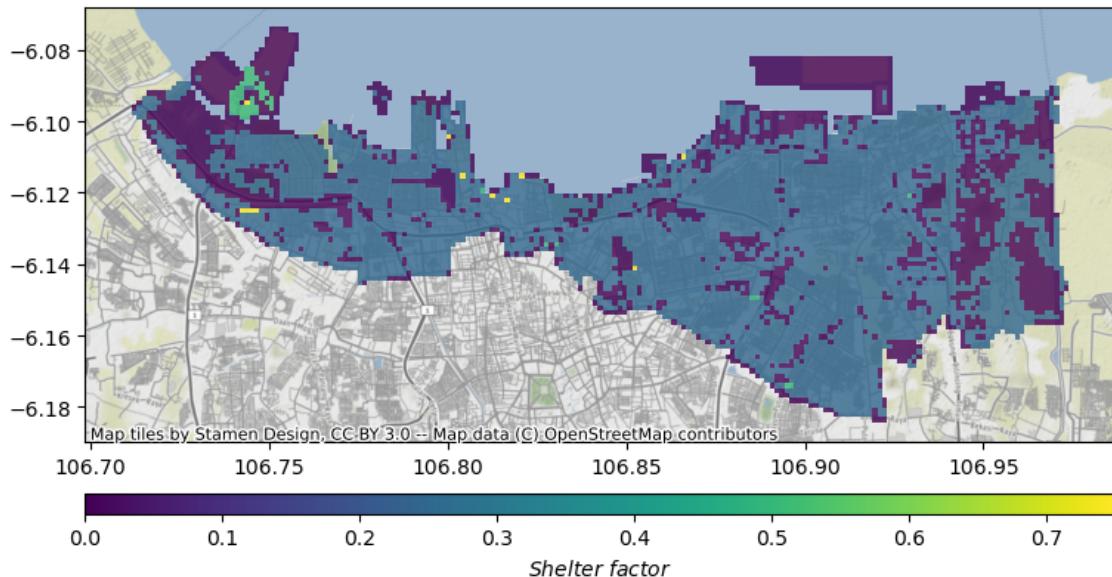


Figure 3.4: Shelter factor map

Sheltering Factor	Type
0	No shelters
0.25	Sparse trees
0.5	Low building
0.75	High building
1.0	Industrial building

Table 3.1: Sheltering factor

3.3.5 No-fly zone layer

The No-fly zone layer refers to a component of a ground risk map or airspace map that delineates areas where drone flights are prohibited. The restrictions in these zones can include areas such as:

- Regulation agency-defined areas, i.e. FAA, and ICAO. They restrict airports and military areas.

- Nature-sensitive areas, such as National Park where flying is restricted.
- Safety and security areas, very crowded public areas, such as the city center, or events.
- Operator-defined areas, where areas are restricted by the operator itself for various reason.

In this paper, the no-fly zone layer is a georeferenced map similar to the obstacle layer. And generally work in the same method to obstacle layers. The No-fly zone layers explicitly prohibit the flight of UAVs by using two different values, -1 if flying is forbidden, and 0 if flying is allowed. The no-fly zone layer is represented by matrix $\mathbf{F}(x, y)$ defined as:

$$\mathbf{F}(x, y) = \begin{cases} -1 & \text{if flight is prohibited.} \\ 0 & \text{if flight is allowed} \end{cases} \quad (3.1)$$

3.4 Ground Risk Assessment

In this research, the risk is described as the potential to cause harm of UAVs to the population on the ground. quantified as the number of fatalities per hour of flight. The risk model is built upon a series of consecutively occurring events, each with its probability. These events are denoted as P_{event} , P_{impact} , and $P_{fatality}$.

$$P_{casualty}(x) = P_{event}(x) \cdot P_{impact}(x) \cdot P_{fatality}(x) \quad (3.2)$$

P_{event} represents the probability of the UAV losing control and descending into the ground. This paper considers four types of events: Ballistic, Unpowered Glide, Flyaway, and Parachute. Mathematical models are utilized to determine the probable impact area for each type of descent. It is important to note that these events are calculated independently of each other. Once the probabilities for each event are computed, they are summed up for each cell, resulting in the probability of the drone impacting that particular cell. This probability is described using a 2D Probability Density Function (PDF). For further information on the descent models, please refer to Sections:3.4.2. The two-dimensional PDF is utilized to calculate the probabilities of P_{impact} and $P_{fatality}$, considering factors such as population density, sheltering factor layer, and impact velocities.

After calculating the UAVs crash location following an uncontrolled descent, we need to calculate the P_{impact} which indicates the likelihood of the UAV strikes individuals. Where the population density and the impact exposure area can have its influence. And finally $P_{fatality}$, which denotes the probability of fatal injuries occurring after a person has been impacted. It depends on the kinetic energy at the time of impact and the sheltering factor.

The overall probability of casualty, $P_{casualty}(x)$, is then computed. The \mathbf{R} is a matrix containing calculated of said $P_{casualty}(x)$ in every element of $\mathbf{R}(x, y)$. This approach al-

lows for the assessment of risk associated with UAS flights by considering various factors and probabilistic models for different events during the flight.

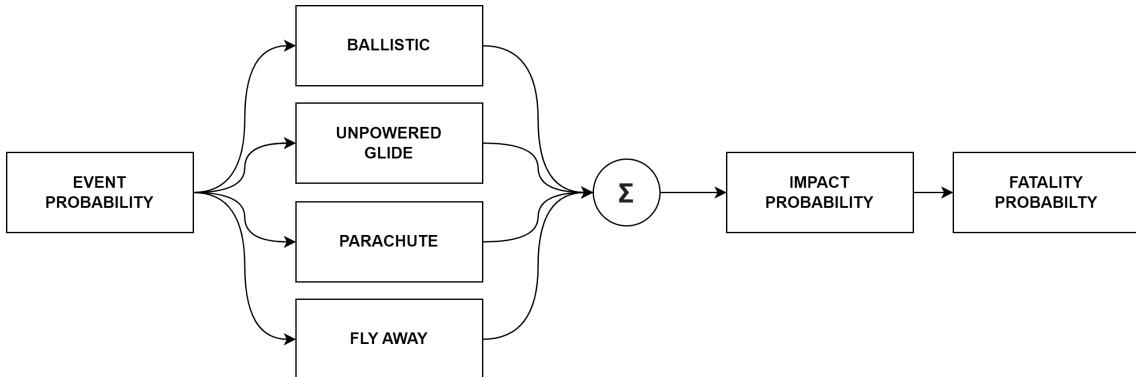


Figure 3.5: General risk assesment flowchart

3.4.1 Failure rate

The failure rate is a metric used to measure the reliability or average failure rate of a system or component. In this study, the MTBF would represent the failure rate. Where MTBF stands for Mean Time Between Failures. MTBF represents the average amount of time that elapses between consecutive failures of a system or component.

MTBF is often expressed in time units, but usually it measures in hours, however, it can also be measured in other such as days or years. A higher MTBF value indicates a longer expected time between failures, suggesting a more reliable system or component.

MTBF is typically calculated using historical failure data. Where this study using the assumption from, Shelley Shelley, 2016 and UAS Task Force “Unmanned Aircraft Systems (UAS) Registration Task Force (RTF) Aviation Rulemaking Committee (ARC)”, 2015 is set into 100 hours.

3.4.2 Descent Event

Information on how the UAV descended into the ground is required for ground risk management. The impact area and the kinetic energy at impact are determined according to the type of descent, which affects how the UAS falls to the ground.

The shape of a UAV has a great influence on how it falls, with fixed wings, generally glides down. While the multirotor fell following the ballistic trajectory. Both of which will influence the falling probability heat map.

These models below are employed to calculate a two-dimensional PDF that quantifies the likelihood of the UAS impacting the ground. The PDF takes into account uncertainties in the initial velocities (horizontal and vertical), drag coefficients, flight altitude, and flight direction.

Furthermore, the model generates a two-dimensional matrix containing the estimated

horizontal and vertical impact velocities. These velocities are instrumental in computing the probabilities of impact (P_{impact}) and fatality ($P_{fatality}$).

3.4.3 Ballistic Descent

The occurrence of ballistic descent arises when the aircraft experiences a catastrophic failure, such as a structural failure that results in the loss of most of its lift. Consequently, the vehicle undergoes a descent solely driven by the forces of gravity and drag.

$$m\dot{v} = m\mathbf{g} - c \cdot A|\mathbf{v}|\mathbf{v} \quad (3.3)$$

The ballistic descent model used in this work is based on the standard second-order drag model. The model takes into account factors such as the mass of the UAV (m), a drag constant (c), area(A), air density(ρ), gravitational acceleration (g), and the vehicle's velocity vector (v). A study done by Cour-Harbo, 2020 provides greater details of the model that also considers uncertainties and variations in wind conditions.

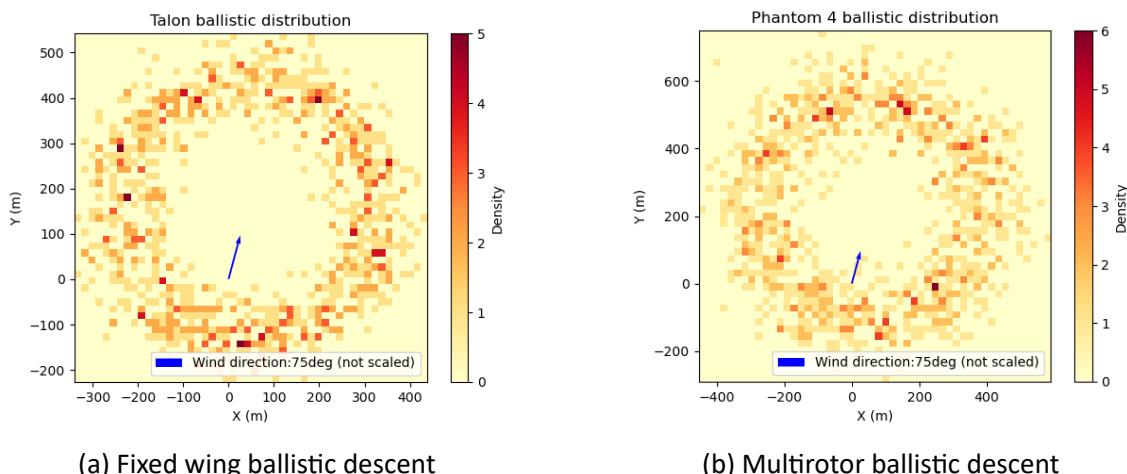


Figure 3.6: Ballistic descent by aircraft type

3.4.4 Glide Descent

Uncontrolled glide may arise depending on the configuration of the aircraft. It is an event when UAV flight profiles are governed solely by the glide ratio or autorotation angle. This event could happen when aircraft loses its power but still retains most of its lift capability.

In the case of a fixed-wing aircraft, this event arises from a loss of thrust or power affecting the flight control surfaces. For a single rotorcraft, it occurs when there is a loss of thrust on the main rotor, causing the aircraft to descend using autopiloted autorotation. However, multi-rotor cannot enter unpowered glide, instead, it enters ballistic descent.

During the uncontrolled glide event, the horizontal distance traveled is computed using the simple formula $dist(h) = \gamma h$, where h represents the flight altitude and γ represents the glide ratio

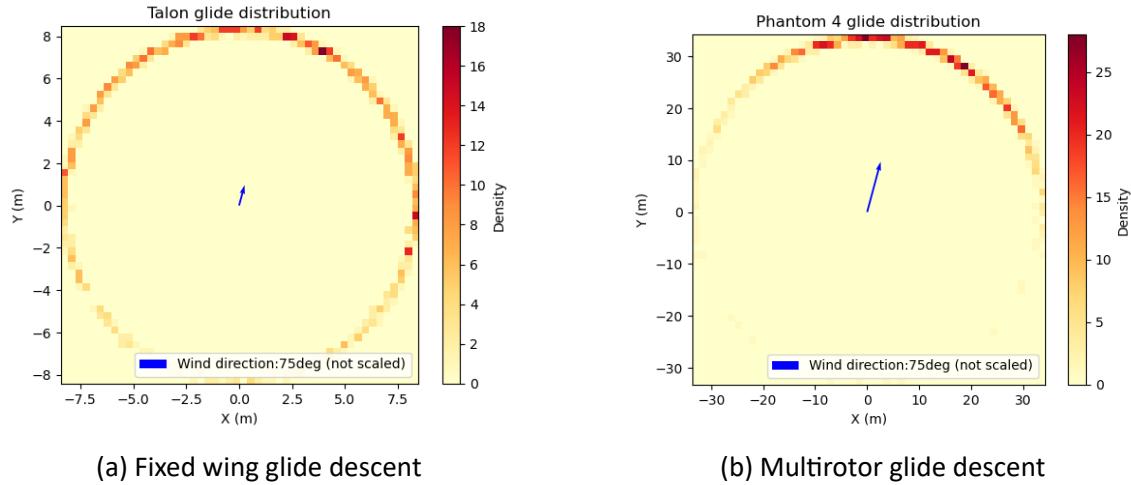


Figure 3.7: Glide descent by aircraft type

3.4.5 Fly away

The fly-away event occurs when the operator loses complete control over the UAV, while the onboard flight control maintains the stability of the vehicle. As a result, the UAS can fly in any direction until it crashes on the ground. This event is modeled based on a reference paper la Cour-Harbo, 2019 , which proposes a model comprising two components. Firstly, the probability of ground impact decreases linearly with the distance from the UAS position until reaching the maximum distance the vehicle can travel. Secondly, the vertical motion of the vehicle is taken into account, with a higher probability of ground impact near the UAS position, modeled as a normal distribution centered around the UAS position. These two contributions are combined linearly using a scaling factor.

In this study, equal probabilities are assumed for both scenarios to be conservative. It is assumed that the fly-away event terminates with an uncontrolled glide descent. Unlike other descent events, the fly-away event can involve motion in all directions. Figure 16 presents the results of the fly-away event.

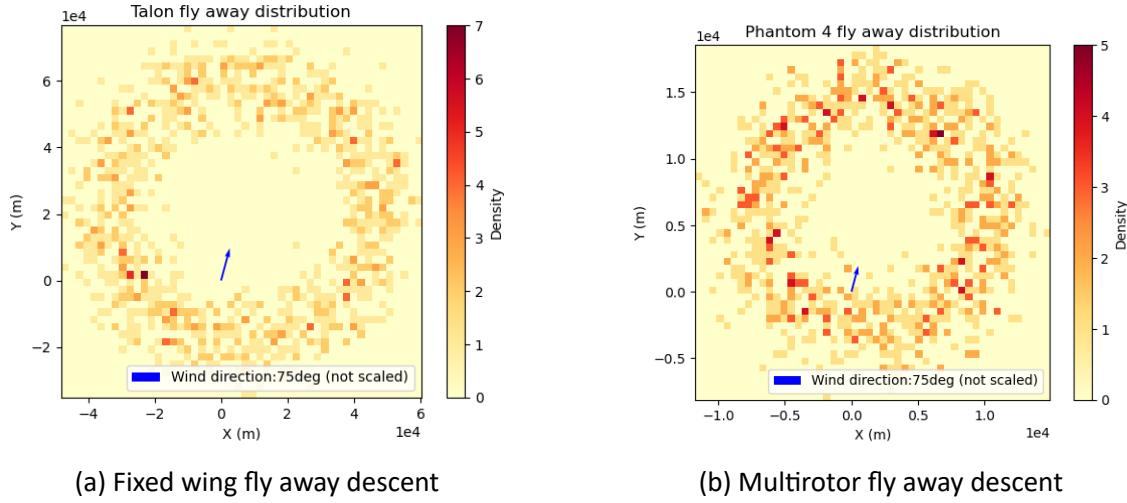


Figure 3.8: Fly away descent by aircraft type

3.4.6 Parachute

The parachute descent starts when the UAV flies downward while the parachute is fully extended. The power plants are typically shut off and the parachute is opened when a problem occurs. After the emergency detection and the parachute fully opening, there is, however, a small delay. The only factors affecting the drop at this point are the parachute's aerodynamic features, which are intended to slow the vertical speed.

To model this event, a methodology described in a referenced paper la Cour-Harbo, 2019 is employed. The authors of the paper consider factors such as the mass of the vehicle, as well as the physical properties of the parachute, including its area and drag coefficient. Notably, the presence of wind significantly alters the descent dynamics, affecting the direction and velocity of the descent.

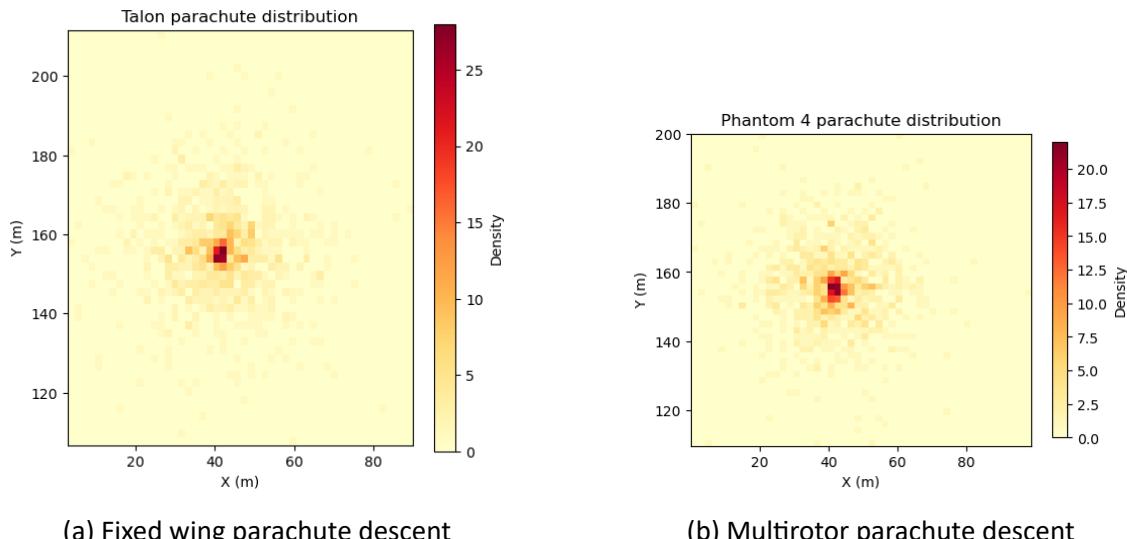


Figure 3.9: Parachute descent by aircraft type

3.4.7 Wind effect

The wind is one of the important variables in creating a ground risk map since wind speed and direction is determined the impact location of each descent event. Where the wind data can be received from weather forecast information before the operation.

Noted that this study assumes and applies constant wind speed and direction when simulating the descent event. This assumption arose when the wind effect usually encompasses a large local area which is often greater than medium to micro-sized UAVs. Added to that, the wind only affected only the descent event and not when the drone is normally operating.

3.4.8 Probability of Impacting an Individual

The P_{impact} is the probability of a UAV striking individual. This parameter is important because, when used in conjunction with the population density parameter previously discussed, it determines the number of people exposed to risk on the ground in the event of an impact.

$$P_{impact}(x, y) = \rho(x, y) \cdot A_{exp} \quad (3.4)$$

The P_{impact} equation describe the population density ρ and exposure area A_{exp}

Similar to sheltering factor, there is no standard approach when calculating the affected area. However, there are two distinct approaches that most researchers find useful, the empirical method and the geometric method.

Empirical methods are the method that uses regression to calculate impact area based on the size or the weight of an aircraft. While the geometric method uses the general dimension of the aircraft to calculate the impact area.

Research done by Melnyk et al., 2014 uses the regression method from Ale and Piers, 2000, based on the accident data of large aircraft to calculate the impacted area. However, this regression is created using aircraft weight of more than 1500kg. This can result in underestimating the lethality of common UAV operations which weigh less than 500kg.

A study done by Aalmoes et al., 2015 proposes two types of solutions. empirical regression and geometric approach. Which stated that, above 500kg the calculation should use the empirical method, or geometric approach if less than 500kg. The empirical data obtained by the Netherland Aerospace Laboratory, NLR, when researching the helicopter crashes behavior, resulted in 3.6 relation. While the geometric approach 3.5 is using human impact model.

$$A_{exp}(m) = 230 \ln\left(\frac{m}{1000}\right) + 330 \quad (3.5)$$

$$A_{exp}(\theta) = 2(r_p + r_{uav}) \frac{h_p}{\tan(\theta)} + \pi(r + r_p)^2 \quad (3.6)$$

Where A_{exp} is in m^2 , with m is kg, θ is impact UAV impact angle. And r_{uav} is UAV radius , h_p is person height, and r_p is person radius all in m,

3.4.9 Probability of Fatality

The probability of fatality, denoted as $P_{fatality}$, represents the likelihood that an impact on a person will result in a fatality. Calculating this probability is challenging due to the various complexities involved in an impact event. The human body reacts differently to different types of impacts, and UAV crashes can involve factors such as blunt impact, lacerations, or chemical explosions caused by the battery. As a result, variables like velocity and mass do not have a simple correlation to the severity of injuries, as the human body's response varies depending on the specific body part affected. This lack of consensus on the calculation method for fatality probability arises from the difficulty of accounting for all the intricacies and variations in impact events. However, it is generally accepted to correlate kinetic impact energy to measure the $P_{fatality}$.

$$P_{fatality} = \frac{1}{1 + e^{-k(E_{imp} - E_{thres})}} \quad (3.7)$$

3.8 is a generalized equation based on the logistic function, where the function maps a value of impact energy E_{imp} to the probability of fatality, with E_{thres} is an energy value that has a probability to cause fatality above 50%. On average, 58 joules of energy can cause a fatality based on the table below.

Information	Value	Unit	Source
Energy for hazardous debris	45	Joule	Cole et al., 1997
Energy required by fragment for 90% probability of fatality	120	Joule	"Hazards to people and aircraft from flight test debris generated at high altitudes Aerospace Sciences Meetings", n.d.
Energy required by fragment to be hazardous to human	58	Joule	"FAA's Approaches to Ground and NAS Separation Distances for Commercial Rocket Launches", n.d.

Table 3.2: Energy correlate to fatality

This study would be using more comprehensive $P_{fatality}$ Equation 3.8, with α as threshold energy to cause 50% probability of fatality in a medium level of shelter $S \geq 0.6$, and β is the threshold energy needed to cause fatality when in a non-sheltered area ($S \leq 0.1$).

$$P_{fatality}(x, y) = \frac{1}{1 + \sqrt{\frac{\alpha}{\beta} \left[\frac{\beta}{E_{imp}(x, y)} \right]^{\frac{1}{4S(x, y)}}}} \quad (3.8)$$

The kinetic energy is computed as:

$$E_{imp}(x, y) = \frac{1}{2}m \cdot v_{imp}(x, y)^2 \quad (3.9)$$

Equation 3.8 is based on Equation 3.8 from a study conducted by Dalamagkidis et al., 2008 and Primatesta et al., 2020. Although it is not accounted for chemical and thermal factors as in Melnyk et al., 2014 and Harwick et al., 2007, we argue that since both Melnyk et al., 2014 and Harwick et al., 2007 only suitable for evaluating per single event and not for generating ground risk map.

3.5 Finding optimal path

After creating a risk map, the next objective is to find the optimal flight path with the least amount of risk. From a naive method point of view, computing flight path is quite easy since 99% of the operation is only a straight path from point A to point B since there is little to no obstacle. Instead, the risk factor has a bigger impact on the Pathfinder compared to real physical obstacles.

There are many types of pathfinder algorithms with each of its advantage, disadvantage, and its most effective usage of operation. In this paper, we would be using A* and Dijkstra's algorithm to find the optimal path and compare each algorithm. There is an explanation in section [A] for reasons why we are not going to use other algorithms.

3.5.1 Problem statement

Given:

- A grid with dimensions $M \times N$, where each cell represents a location in the grid.
- A start cell S and a goal cell G in the grid.
- Each cell in the grid has an associated risk factor (a non-negative real value) that represents the level of risk or danger associated with that cell.

Objective: Find the shortest path from the start cell S to the goal cell G that minimizes the cumulative risk factor along the path.

Formally, let::

- Let $G = (V, E)$ be the graph representation of the grid, where V represents the set of cells and E represents the set of edges connecting adjacent cells in the grid.
- Let $w(u, v)$ be the weight or cost associated with an edge (u, v) in E . In this case, the weight $w(u, v)$ represents the risk factor of moving from cell u to cell v .
- Let $d(u)$ represent the shortest path distance from the start cell S to cell u .

The problem is to find the shortest path $P = (S, v_1, v_2, \dots, G)$ that minimizes the cumulative risk factor along the path, which can be represented by the sum of the weights along the path:

$$\min \sum w(v_i, v_{i+1}) \quad (3.10)$$

subject to:

- P is a valid non negative weighted path from S to G in the graph G .
- $d(G)$ is minimized.

The objective is to find the path P that minimizes the cumulative risk factor, taking into account the risk factors associated with each cell in the grid.

This problem can be solved using various algorithms such as Dijkstra's algorithm, A*, or even extensions like Risk-Aware A* that consider risk factors in the pathfinding process.

Noted that the specific formulation and constraints of the problem may vary depending on additional considerations such as movement constraints, specific risk factor calculations, and other domain-specific requirements. The above problem statement provides a general framework for finding the shortest path based on risk factor in a grid system.

Graph Before using any pathfinder algorithm, the grid map must be defined in the graph. In computer science and mathematics, a graph is a data structure that represents a collection of interconnected nodes, called vertices, (V), and the relationships between them, called edges, (E).

A graph consists of two main components: Vertices (Nodes): These are the fundamental units of a graph and represent entities or objects. Each vertex is typically labeled with a unique identifier or value. Edges: These represent the connections or relationships between vertices. An edge connects two vertices and can be either directed or undirected.

A directed edge has an orientation or a specific direction, while an undirected edge does not. Edges can also have weights or costs associated with them to represent the strength of the relationship or distance between vertices.

A grid typically falls into the category of a regular graph or a lattice graph. A regular graph is a graph in which every vertex has the same number of neighbors. In a grid, each cell is typically connected to its adjacent cells, resulting in a regular graph structure. The number of neighbors depends on the dimensionality of the grid.

The grid system, with its regular arrangement of cells and connectivity between adjacent cells, can also be considered a form of lattice graph. A lattice graph is a specific type of regular graph that forms a lattice-like structure. A lattice graph consists of a regular arrangement of vertices in a grid-like pattern, where each vertex is connected to its adjacent vertices.

3.5.2 Dijkstra's algorithm

Dijkstra's algorithm is a well-known and widely used algorithm for finding the shortest path between a given source node and all other nodes in a graph. It guarantees optimality, meaning it always finds the shortest path, as long as the graph has non-negative edge weights.

Dijkstra's algorithm uses a cost function to calculate the shortest path from a given start node to all other nodes in a graph. The distance in Djikstra's case is represented as a cost function of $g(x)$.

$$g(x) = \min(g(x_{goal}), g(x_n)) \quad (3.11)$$

The function $g(x_n)$ represents the shortest distance from the start node to a given node x_n in the graph. It is initialized to a very large value, often infinity, for all nodes except the start node, which is assigned a distance of 0.

$$g(x_n) = \sum_{i=1}^{n-1} g(x_{i-1}) + c(x_{n-1}, x_n) \quad (3.12)$$

where, $\sum_{i=1}^{n-1} g(x_{i-1})$ is all combined cost from a start node to the current node, and $c(x_{n-1}, x_n)$ is a risk-weighted cost function, which describes to

$$c(x_{n-1}, x_n) = \frac{r(x_{n-1}) + r(x_n)}{2} \cdot d(x_{n-1}, x_n) \quad (3.13)$$

where the risk r_x is the average between two nodes, and $d(x_{n-1}, x_n)$ is an euclidean distance between two nodes.

The algorithm iteratively selects the node with the minimum distance among the unvisited nodes and updates the distances of its neighboring nodes if a shorter path is found. This process continues until all nodes have been visited or until the goal node is reached, depending on the specific problem requirements.

By continuously updating the distances of nodes based on the weights of the edges, Dijkstra's algorithm ensures that it explores paths with lower total costs, gradually converging towards the shortest path from the start node to each node in the graph.

3.5.3 A* algorithm

A* algorithm is an extension of the Dijkstra algorithm and uses a heuristic function to guide the search. From 3.14 we can see that the addition of heuristic function $h(x_n)$. Noted that the implementation of $g(x)$ is similar to previously mentioned in 3.12.

The cost function can be represented as:

$$f(x) = g(x_n) + h(x_n) \quad (3.14)$$

where:

- $f(n)$ is the total estimated cost of node n,
- $g(n)$ is the actual cost from the start node to node n,
- $h(n)$ is the heuristic estimate of the remaining cost from node n to the goal node.

The cost function in the A* algorithm typically consists of two components: the actual cost to reach a vertex from the start node, and the heuristic estimate of the remaining cost from the vertex to the goal node. The sum of these two components represents the total estimated cost from the start node to the goal node through a particular vertex.

The heuristic estimate, $h(x_n)$, must be an admissible (underestimating) heuristic function that provides an estimate of the remaining cost from node n to the goal node. Admissibility is a requirement for the algorithm to be able to look for the best solution in the graph. If admissibility is not implemented, overestimation might occur in the algorithm, and it may overlook nodes that would lead to the optimal solution.

The solution for admissible heuristic implementation is based upon the paper from Primatesta et al., 2019. Where the author *dialed down* the heuristic value by multiplying it by the risk cost between x_{goal-1} and the x_{goal} itself, this is to ensure conservative heuristic admissibility.

$$\begin{aligned} h(x_n) &= \frac{r(x_n) + r_{min}}{2} \cdot \text{dist}(x_n, x_{n+1}) + \text{dist}(x_{n+1}, x_{goal-1})r_{min} \\ &\quad + \frac{r(x_{goal}) + r_{min}}{2} \cdot \text{dist}(x_{goal-1}, x_{goal}) \end{aligned} \quad (3.15)$$

Algorithm 1 Dijkstra's Algorithm

```
1: function DIJKSTRA(start, goal)
2:   distances  $\leftarrow \{\}$ 
3:   previous  $\leftarrow \{\}$ 
4:   for all node in graph do
5:     distances[node]  $\leftarrow \infty$ 
6:     previous[node]  $\leftarrow$  undefined
7:   end for
8:   distances[start]  $\leftarrow 0$ 
9:   queue  $\leftarrow$  PriorityQueue()
10:  queue.insert(start, 0)
11:  while queue is not empty do
12:    current  $\leftarrow$  queue.pop_min()
13:    if current is goal then
14:      return ReconstructPath(previous, current)
15:    end if
16:    for all neighbor of current do
17:      distance  $\leftarrow$  distances[current] + cost(current, neighbor)
18:      if distance  $<$  distances[neighbor] then
19:        distances[neighbor]  $\leftarrow$  distance
20:        previous[neighbor]  $\leftarrow$  current
21:        queue.insert(neighbor, distance)
22:      end if
23:    end for
24:  end while
25:  return Failure
26: end function

27: function RECONSTRUCTPATH(previous, current)
28:   path  $\leftarrow$  [current]
29:   while previous[current] is defined do
30:     current  $\leftarrow$  previous[current]
31:     path.prepend(current)
32:   end while
33:   return path
34: end function
```

By summing the actual cost and the heuristic estimate, the A* algorithm prioritizes nodes with lower total costs, resulting in a search that tends to explore paths that are likely to be closer to the goal. This heuristic-guided exploration helps to efficiently find the shortest path from the start node to the goal node.

To summarize, when there is a single unique optimal path, A* can discover it more efficiently than Dijkstra's algorithm by utilizing an admissible heuristic function. While Dijkstra's algorithm has the capability to eventually find the absolute optimal path given sufficient time. By comparing the results of A* and Dijkstra's algorithm, we can assess the performance of A*.

Algorithm 2 A* Algorithm

```
1: function ASTAR(start, goal)
2:   openSet  $\leftarrow \{start\}$ 
3:   closedSet  $\leftarrow \{\}$ 
4:   cameFrom  $\leftarrow \{\}$ 
5:   gScore  $\leftarrow \{\}$ 
6:   hScore  $\leftarrow \{\}$ 
7:   fScore  $\leftarrow \{\}$ 
8:   gScore[start]  $\leftarrow 0$ 
9:   hScore[start]  $\leftarrow \text{HEURISTIC}(start, goal)$ 
10:  fScore[start]  $\leftarrow gScore[start] + hScore[start]$ 
11:  while openSet is not empty do
12:    current  $\leftarrow$  node in openSet with the lowest fScore
13:    if current is goal then
14:      return RECONSTRUCTPATH(cameFrom, current)
15:    end if
16:    openSet  $\leftarrow$  openSet \ {current}
17:    closedSet  $\leftarrow$  closedSet  $\cup \{current\}$ 
18:    for all neighbor in NEIGHBORS(current) do
19:      if neighbor in closedSet then
20:        continue
21:      end if
22:      tentativeGScore  $\leftarrow gScore[current] + \text{cost}(current, neighbor)$ 
23:      if neighbor not in openSet or tentativeGScore < gScore[neighbor] then
24:        cameFrom[neighbor]  $\leftarrow$  current
25:        gScore[neighbor]  $\leftarrow$  tentativeGScore
26:        hScore[neighbor]  $\leftarrow \text{HEURISTIC}(neighbor, goal)$ 
27:        fScore[neighbor]  $\leftarrow gScore[neighbor] + hScore[neighbor]$ 
28:        if neighbor not in openSet then
29:          openSet  $\leftarrow$  openSet  $\cup \{neighbor\}$ 
30:        end if
31:      end if
32:    end for
33:  end while
34:  return Failure
35: end function

36: function RECONSTRUCTPATH(cameFrom, current)
37:   path  $\leftarrow [current]$ 
38:   while current in cameFrom do
39:     current  $\leftarrow$  cameFrom[current]
40:     PREPEND(current, path)
41:   end while
42:   return path
43: end function
```

3.6 UPX, UAV Pathfinding eXtension

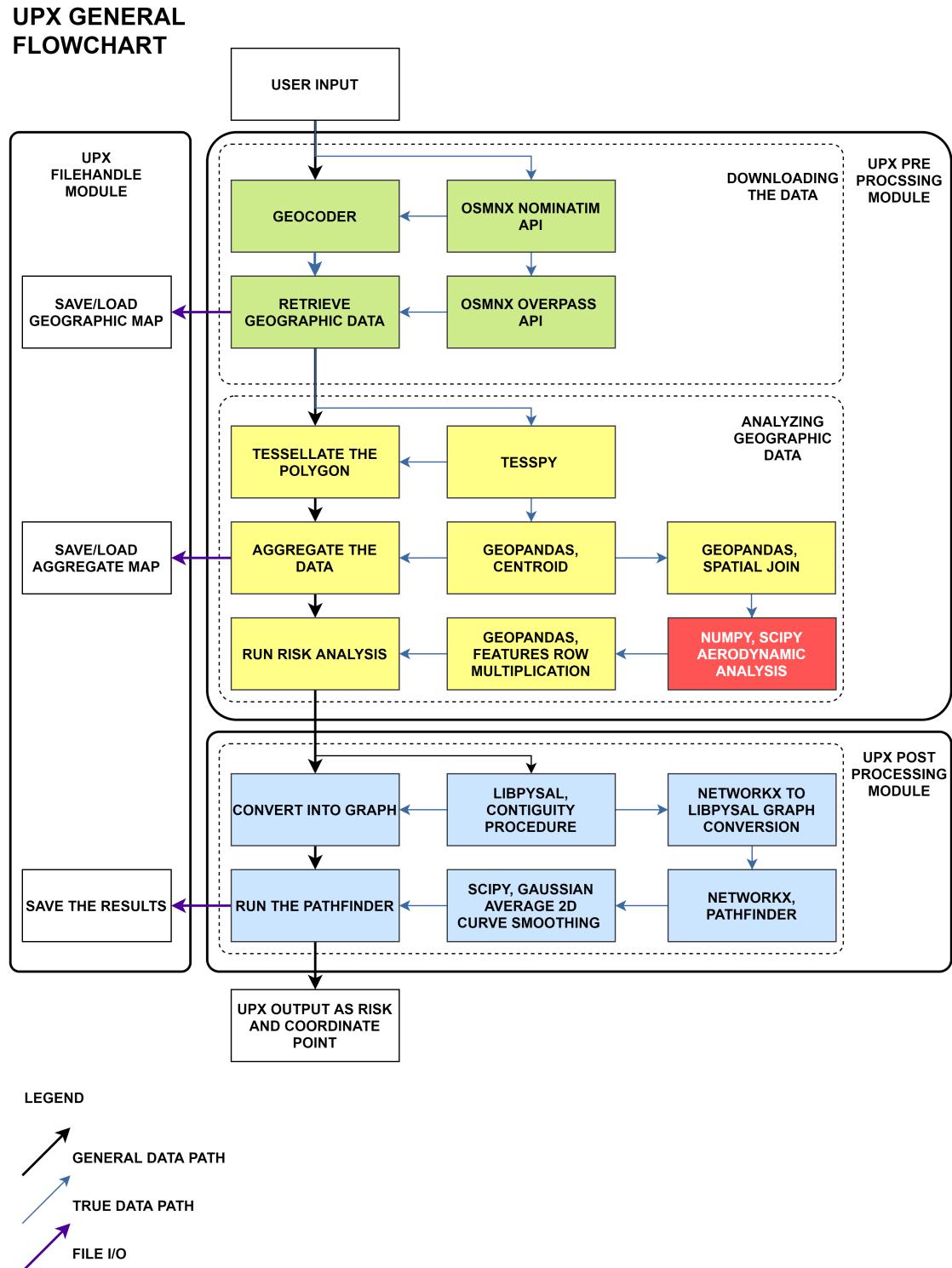


Figure 3.10: UPX architecture design

UAV Pathfinding eXtension, UPX, is both a python libraries and application to find the optimal risk-informed flight path. In general, UPX works by retrieving geographic data from OSM, receiving geographic data, then being discretized and aggregated. The risk map is then converted into a graph for the pathfinding algorithm. The risk mentioned Risk in UPX is rules to determine the probability of fatality in case the UAV happens to malfunction and falls in that area.

UPX was built based on the concern of negative consequences of increase frequency of UAVs flying in urban areas. While UPX is intended to be used to aid UAV operation, its library can be used in any type of aircraft as long they provide aerodynamic parameters and aircraft performance data. However, some caveats would be explained in the UPX limitation below.

3.6.1 Limitation

Firstly, UPX is strictly for a preliminary assessment only. It does not handle real-world phenomena such as sudden wind gale, rain, storm, or other aircraft that, unfortunately, are present in the path of the UAV. For that reason only, we hope the pilot practice safe flying and the UAV itself is provided with the failsafe feature.

3.7 UPX Python and packages

3.7.1 Python

Python is an interpreted programming language. Interpreted language has the benefit of being an easier language to compare to a compiled language. Added to this, it has massive scientific-related packages and libraries. Combine with quick deployment and its STEM library, it is a perfect choice for engineering, mathematics, and physics analysis. Based on previous criteria it is a great choice for creating UAV path planner applications and libraries.

While Python indeed has a great advantage. Unfortunately, its comparatively easy language came with a great price. An interpreted language is inherently slow, 8 to 100 times slower compared to C/C++ language. Base Python language also cannot use more than one thread because its GIL, Global Interpreter Lock, prevents the threads from sharing memories and create a parallel workload. Fortunately, most of the heavy calculation is handled by the c/c++ library, where Python uses the caller function to launch the libraries.

3.7.2 GeoPandas

GeoPandas is one of the GIS libraries built for tabulating and analyzing geographical spatial analysis in python. GeoPandas itself can be regarded as the extension of Pandas library, where Pandas is general tabular data processor. GeoPandas has geographic spatial and attribute related function such as overlay, spatial join, clipping to group, aggregate, and measure geospatial data.

GeoPandas exports and imports files in standard GIS-supported files such as GeoJSON, the file, and the Shapely object, to name a few. In Python GeoPandas tabular data is called GeoDataFrame, and GeoSeries. What differentiates between GeoDataFrame and regular DataFrame is the addition of a geometry column. The geometry column can contain Multi/Polygon, Multi/Linestring, and Point shapes.

3.7.3 Shapely

Shapely is a geometry calculation and manipulation packages available for python. Many of geospatial function in GeoPandas relies on Shapely itself. Overlay, clipping, spatial binary combination in GeoPandas can be found in Shapely. In this paper Shapely is being used to create custom filter for heat map function of aircraft falling probability.

3.7.4 Tesspy

Tesspy, a tesselation library for Python. Its purpose, as the name implies, is to create a tessellated polygon. Tesspy can be used as long as the polygon object is derived from the Shapely object. Tesspy offers type of tessellated map that commonly use for geospatial processing. Square grid, hexbin, Voronoi city block, and adaptive square is the type of tessellated maps, to name a few.

3.7.5 OSMnx

OSMnx is a combination of OSM and NetworkX. OSMnx is mainly being used to create graph representation of the road network. It uses the GeoPandas data frame, to build the network, as long as the roads in question are connected at least with another road. But for this paper, OSMnx is only being used as a geocoder and reverse geocoder to retrieve GeoPandas data.

3.7.6 NetworkX

To create a graph structure, NetworkX provides the necessary function to do that. NetworkX offers graph creation, graph traversal algorithms, and graph manipulation. The graph, as previously mentioned, in a combination of nodes and edges. The A* pathfinder for the UAV Path finder also uses the NetworkX built-in A*.

CHAPTER 4

RESULTS

4.1 Ground risk map generation

4.1.1 UAVs Specification

In this study, we are going to analyze the risk factor of multirotor and fixed-wing UAVs. In comparison, Talon and Parrot Disco are fixed-wing. While DJI Phantom 4 and DJI Inspire 2 are multirotor. Four of them are chosen since they are generally known as a go-to standard in the commercial drone industry. Each category is also differentiated into two different sizes, where Talon and Inspire 2 are the large size, while Phantom 4 and Disco are much smaller UAVs. All properties are obtained from manufacturer design specifications.

As for the aerodynamic properties of these drones, we are using a very simple drag model. This will not include an Induced drag since we cannot predict every possible angle of attack. It is also beneficial to use simplify drag model since it would give a higher impact velocity, which increases the impact energy and results in a much more conservative value.

It is worth mentioning the mechanic of the glide model in this risk analyzer. Since only fixed wings are capable of gliding, and to an extent, a variable-pitch helicopter by autorotation, it is almost impossible to make a glide descent on a fixed-pitch propeller multirotor. Where for the multirotor, ballistic descent is the more likely outcome, even though there is little to no damage in the aircraft. It is then assumed that a glide descent, and to an extent a glide speed in multirotor, as a low powered horizontal descent where the power is barely sustaining a hover maneuver

Specification	Talon	Parrot Disco	Phantom 4	Inspire 2
Type	Fixed wing	Fixed wing	Quadcopter	Quadcopter
Mass (kg)	3.75	0.75	1.4	4.25
Front area (m^2)	0.1	0.07	0.02	0.04
Radius (m)	0.88	0.575	0.2	0.4
Maximum flight time	1.25 h	45 min	20 min	25 min
Ballistic drag coefficient	N(0.9, 0.2)	N(0.9, 0.2)	N(0.7, 0.2)	N(0.7, 0.2)
Horizontal initial speed (m/s)	N(18, 2.5)	N(15, 2.5)	U(0, 15)	U(0, 20)
Vertical initial speed (m/s)	0	0	0	0
Glide speed (m/s)	16	12	7.5	10
Glide ratio	N(12, 2)	N(2.7, 0.8)	N(2.7, 0.8)	N(2.7, 0.8)
Parachute drag coefficient	N(13, 0.2)	N(0.9, 0.2)	N(0.9, 0.2)	N(0.9, 0.2)
Parachute area	1	0.5	0.5	0.5

Table 4.1: UAVs Specifications

4.2 Objective

The ground risk would be performed in North Jakarta, Indonesia. The place is chosen for its dense population, an assortment of building heights, large water bodies, and multiple residential and commercial zone areas. Shown in 4.1

And for the operation itself, the UAV would be flying from Pluit Village to Jakarta Sailing Vocational School. This objective is chosen based on the criteria previously mentioned.

We would compare the performance of the Risk A* Algorithm vs the Normal A* algorithm. We do not include Djikstra's algorithm in this section since the A* algorithm provided by the NetworkX library is already admissible and able to obtain optimal pathways.

It is worth mentioning the normal A* is the same for all types of aircraft since aircraft properties are only important in Risk A* in generating various descent PDFs.

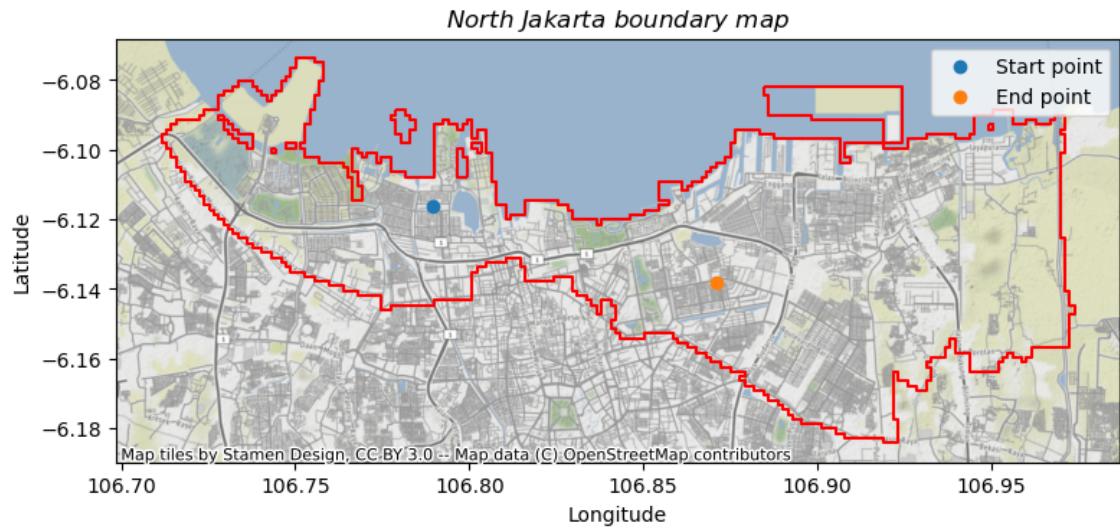


Figure 4.1: Operation objective

4.3 PDF risk map

4.3.1 Talon

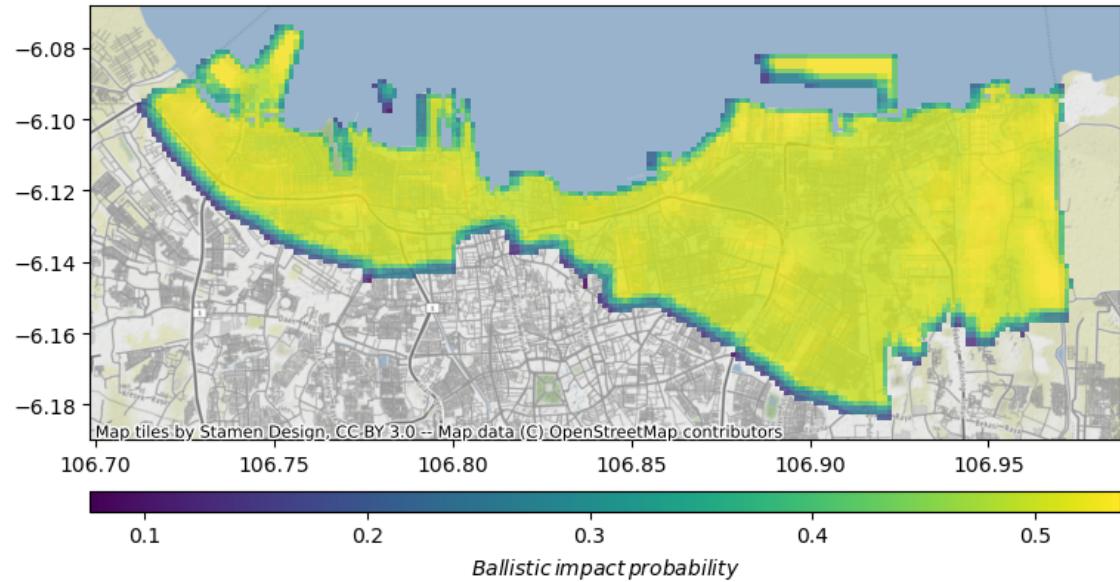


Figure 4.2: Talon ballistic PDF

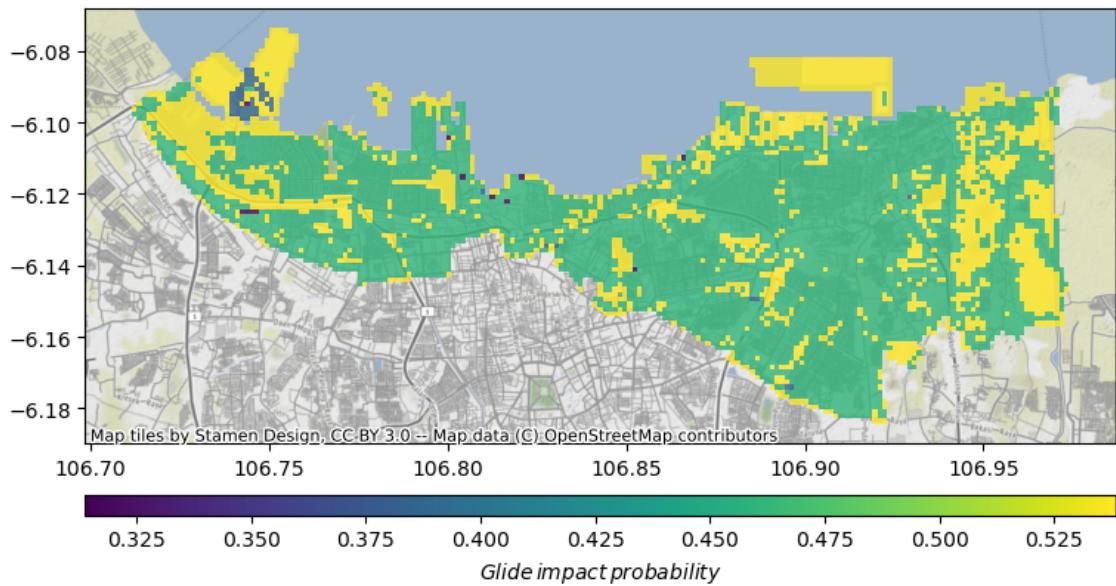


Figure 4.3: Talon glide PDF

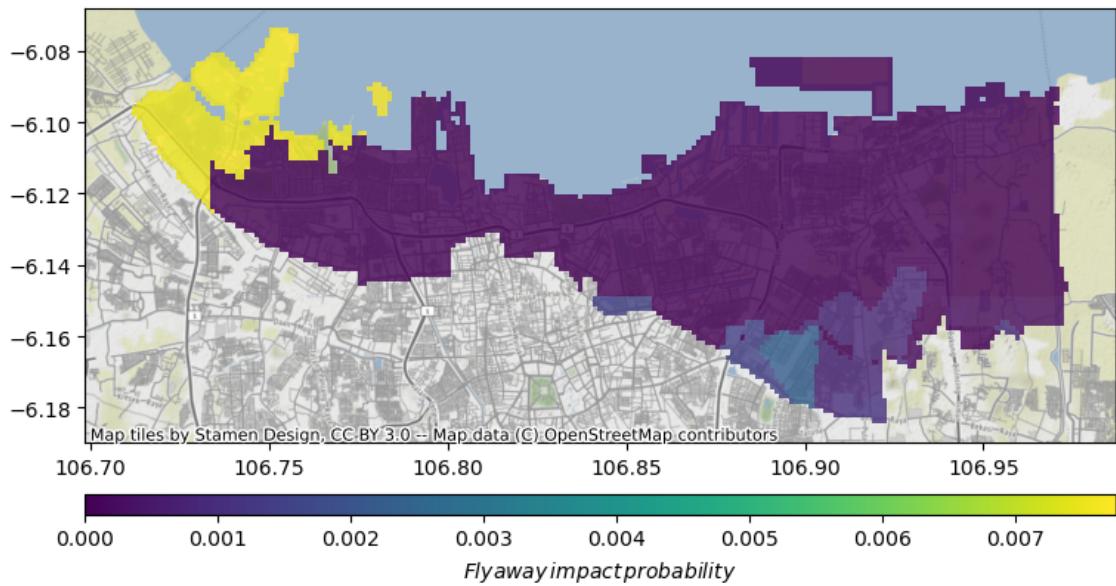


Figure 4.4: Talon fly away PDF

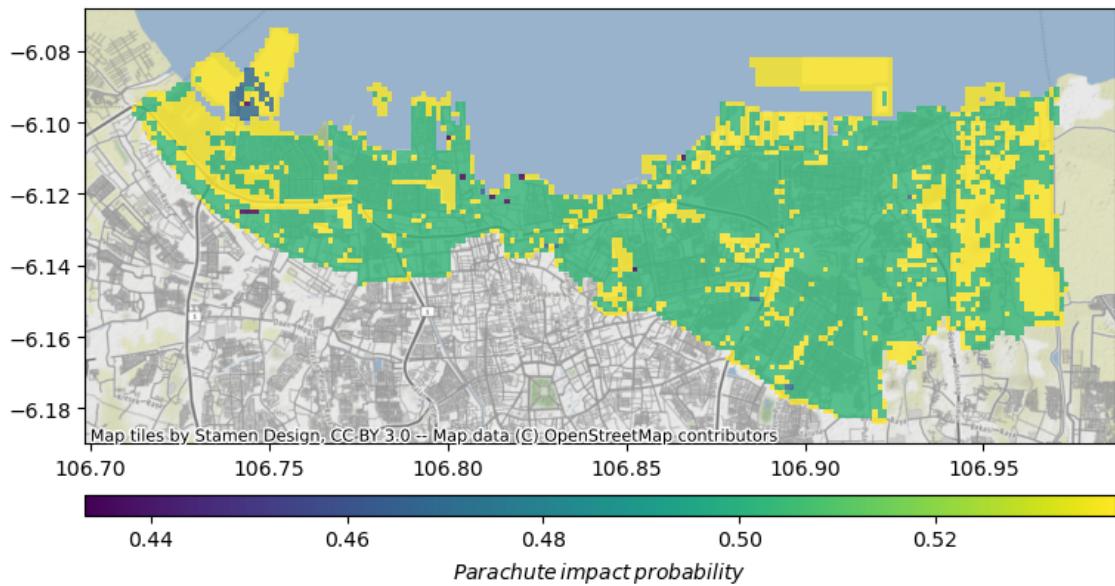


Figure 4.5: Talon parachute PDF

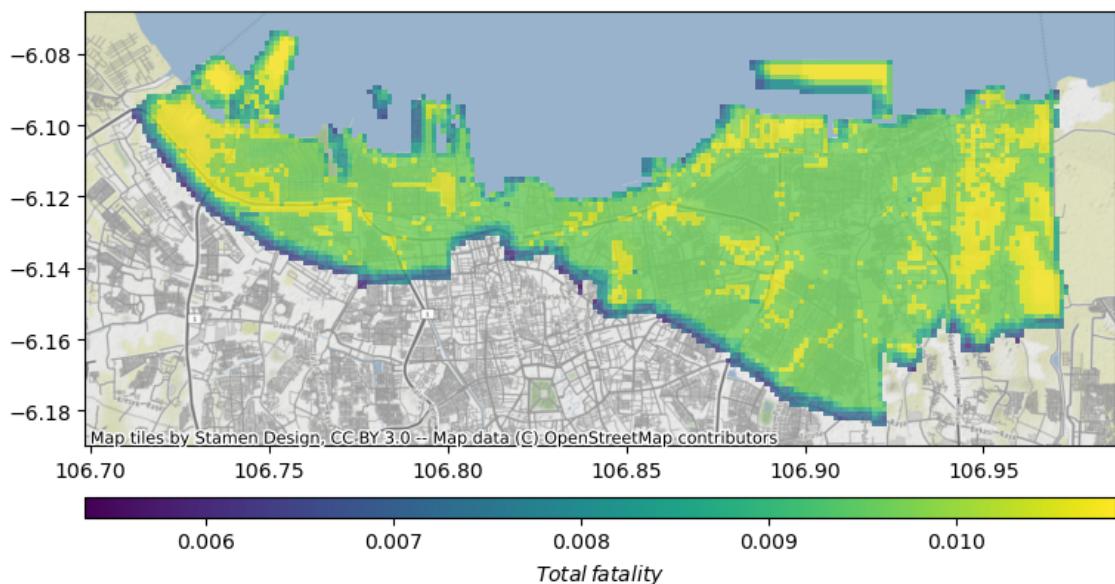


Figure 4.6: Talon fatality risk PDF

4.3.2 Parrot Disco

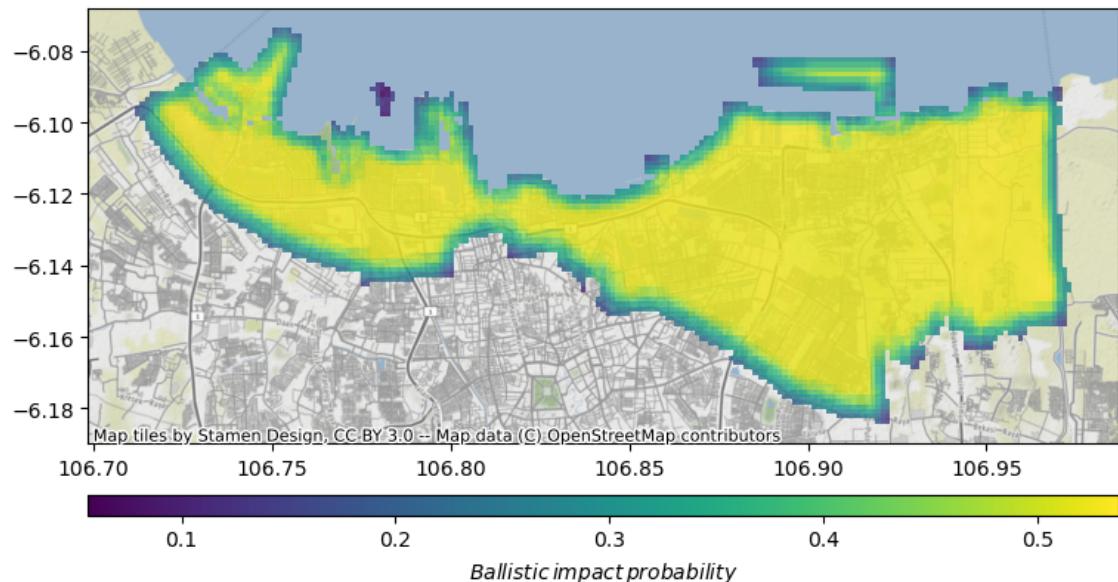


Figure 4.7: Disco ballistic PDF

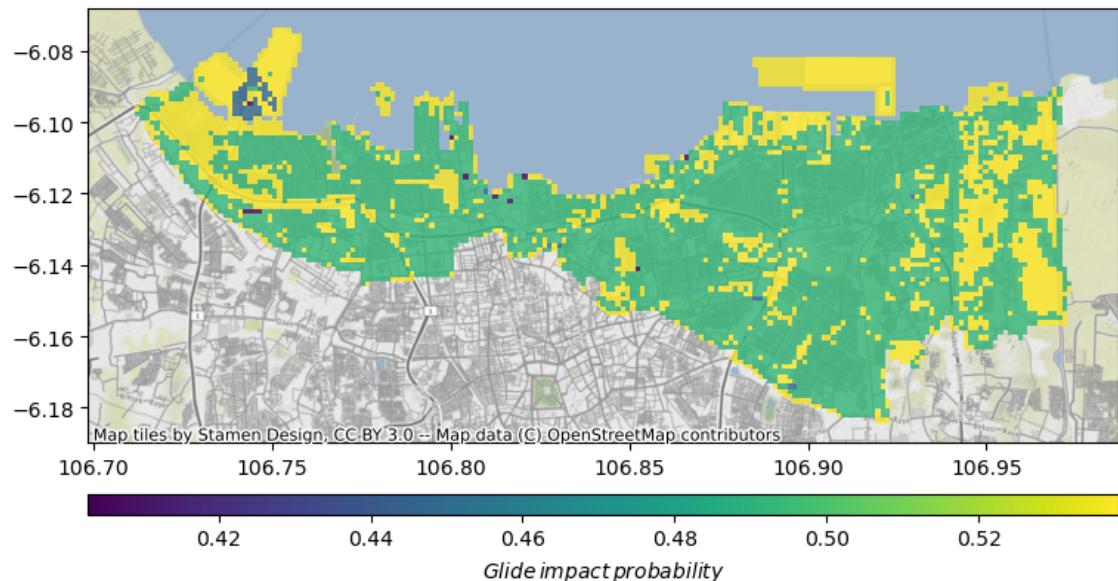


Figure 4.8: Disco glide PDF

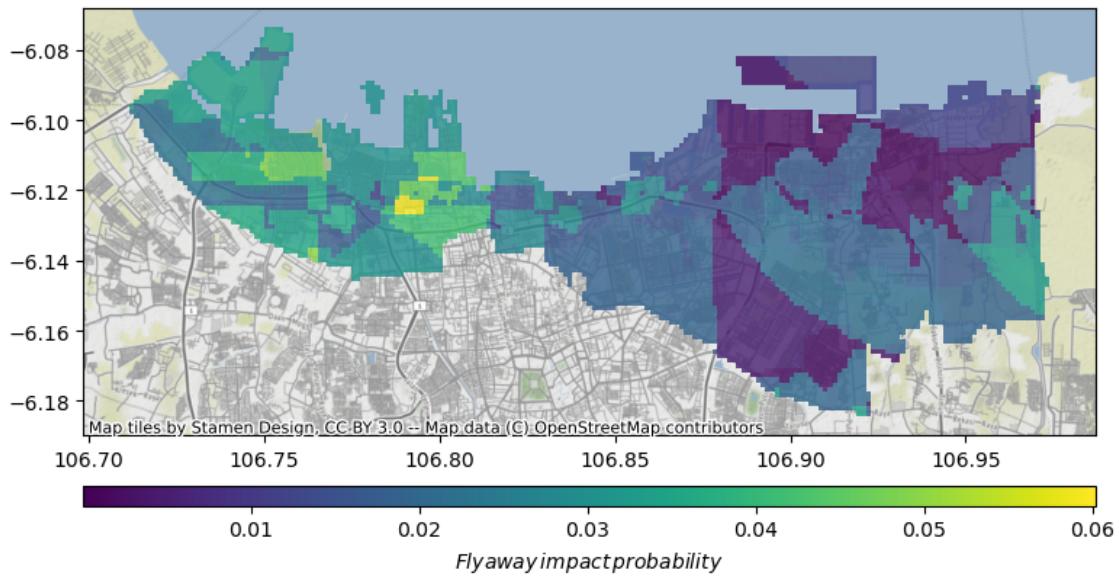


Figure 4.9: Disco fly away PDF

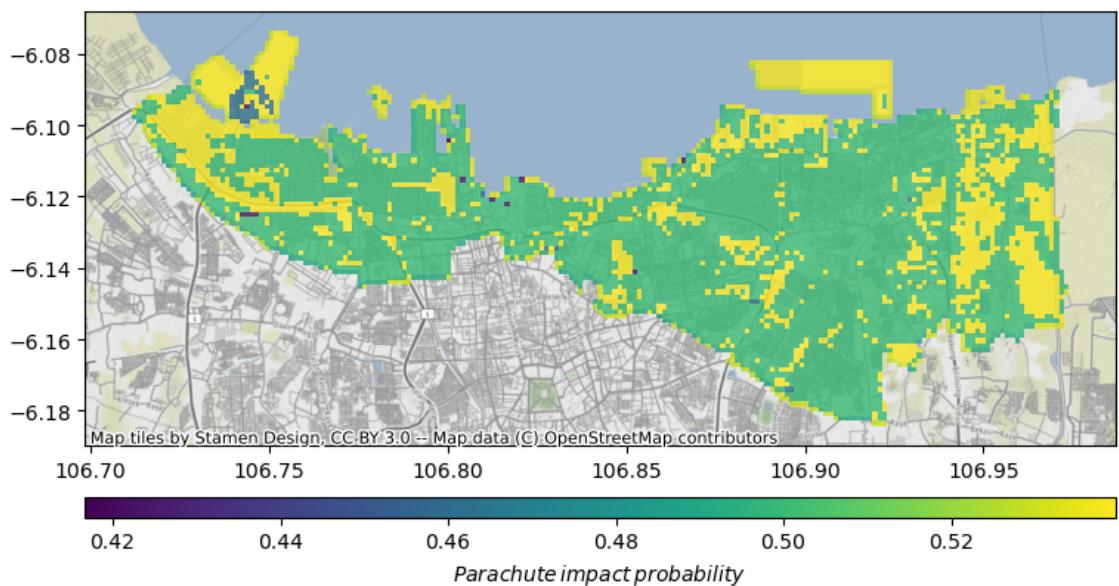


Figure 4.10: Disco parachute PDF

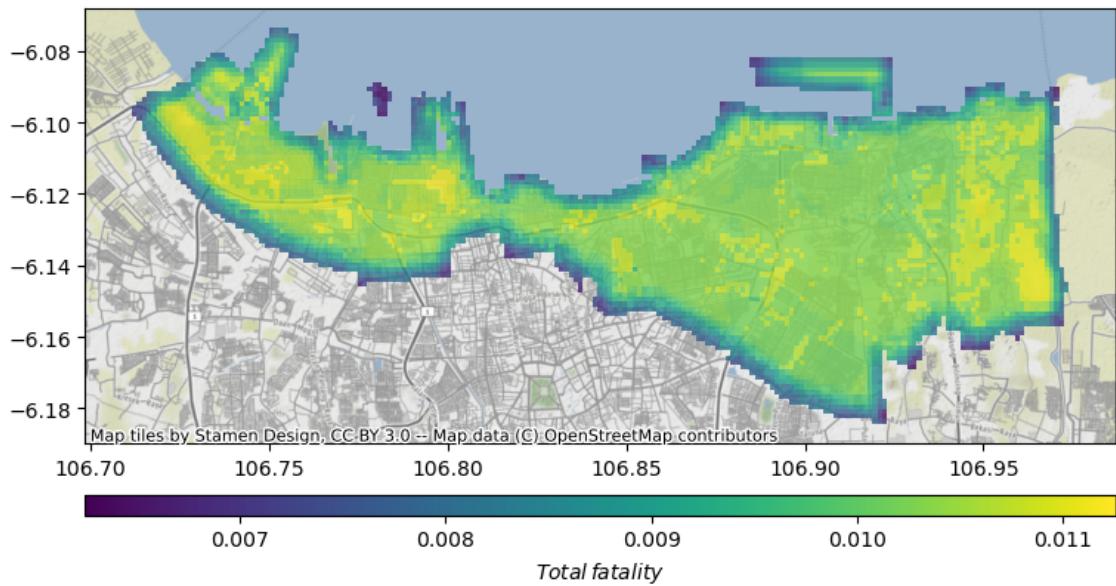


Figure 4.11: Disco fatality risk PDF

4.3.3 DJI Phantom 4

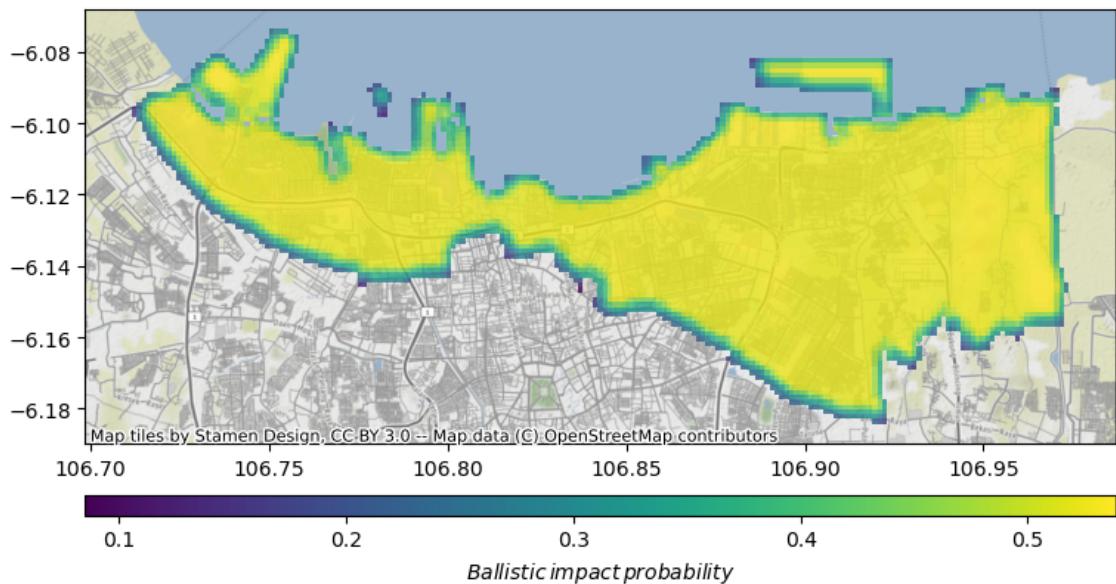


Figure 4.12: DJI Phantom 4 ballistic PDF

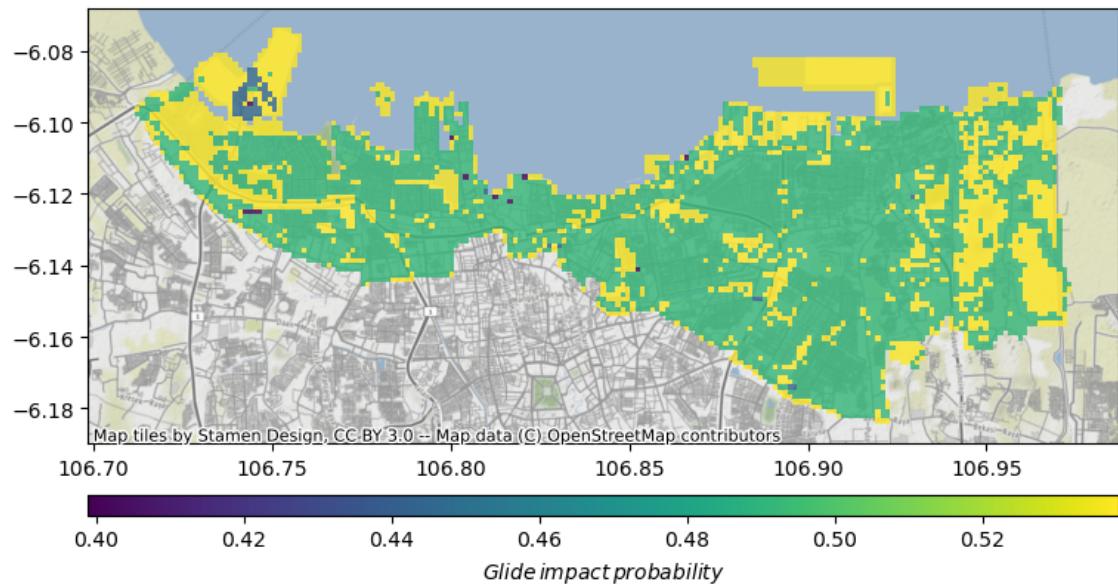


Figure 4.13: DJI Phantom 4 glide PDF

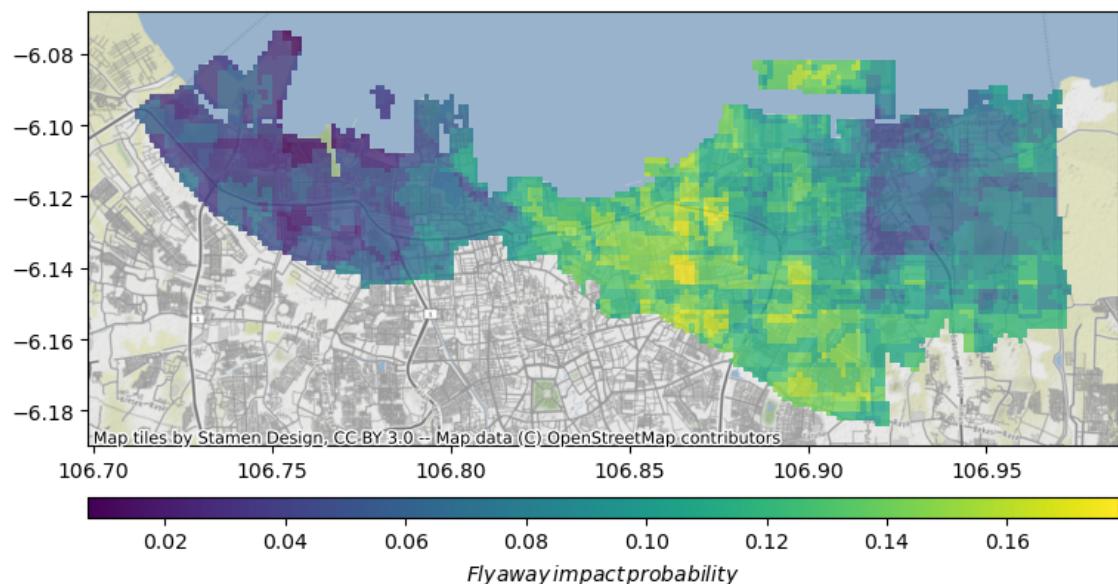


Figure 4.14: DJI Phantom 4 fly away PDF

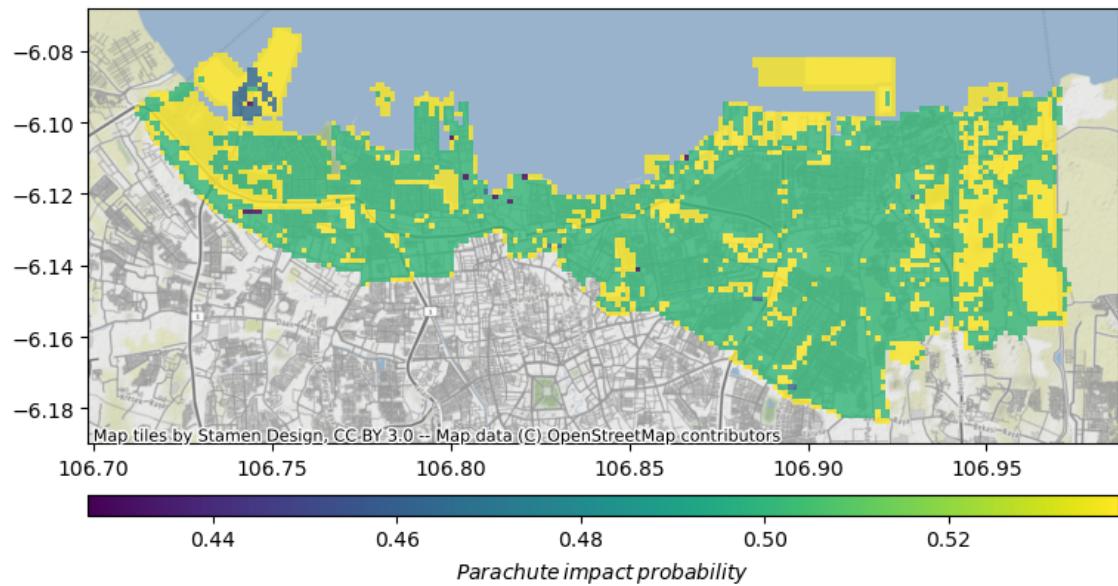


Figure 4.15: DJI Phantom 4 parachute PDF

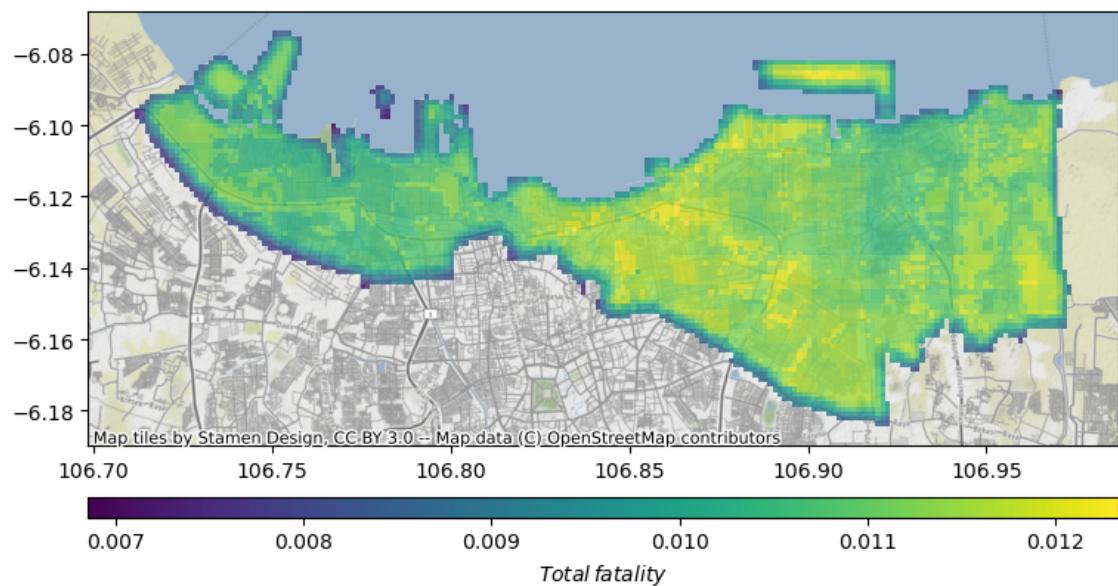


Figure 4.16: DJI Phantom 4 fatality risk PDF

4.3.4 DJI Inspire 2

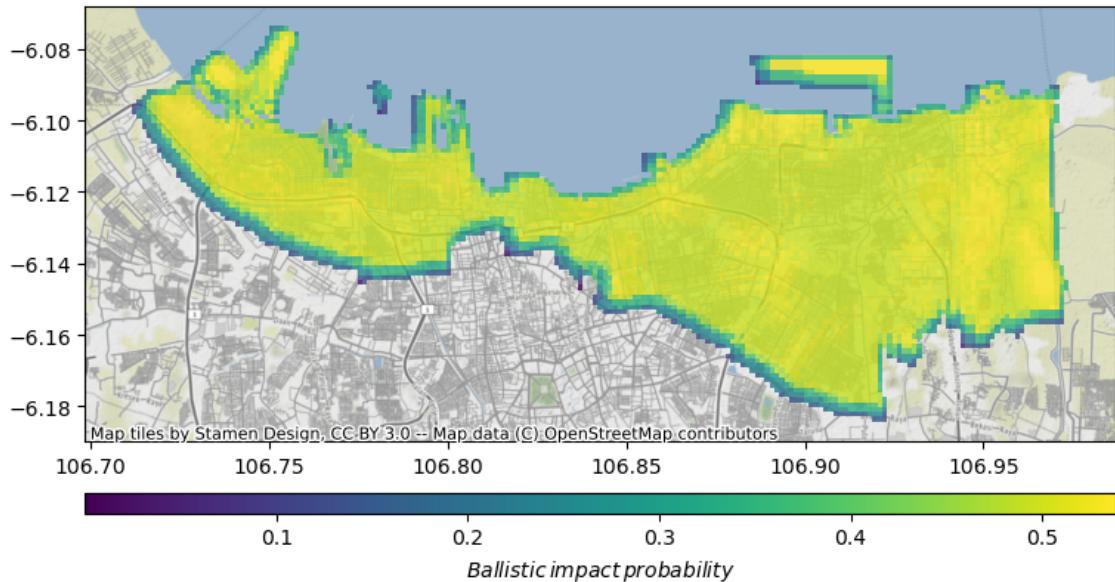


Figure 4.17: DJI Inspire 2 ballistic PDF

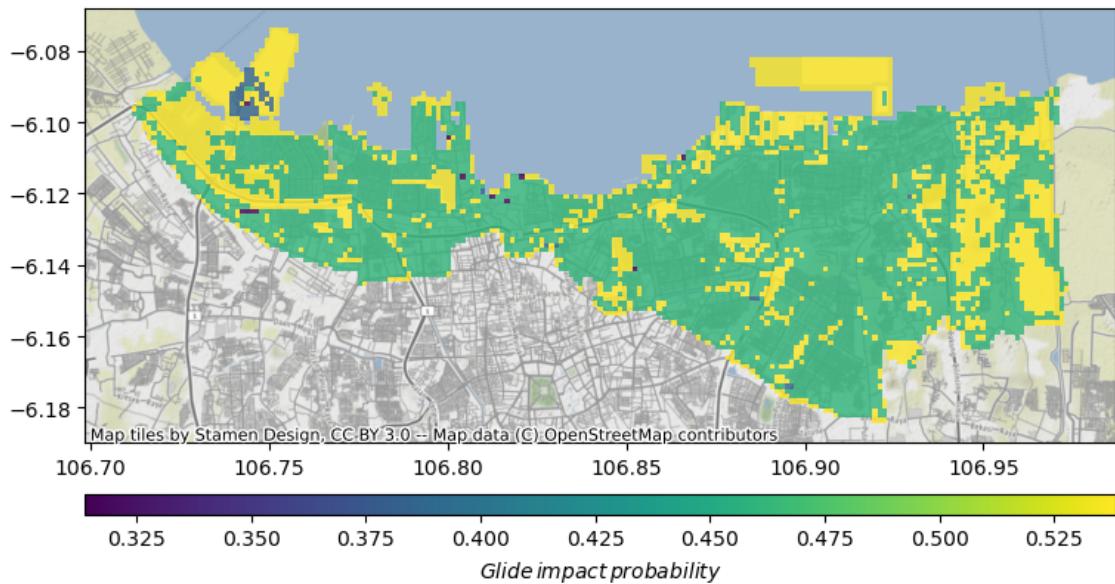


Figure 4.18: DJI Inspire 2 glide PDF

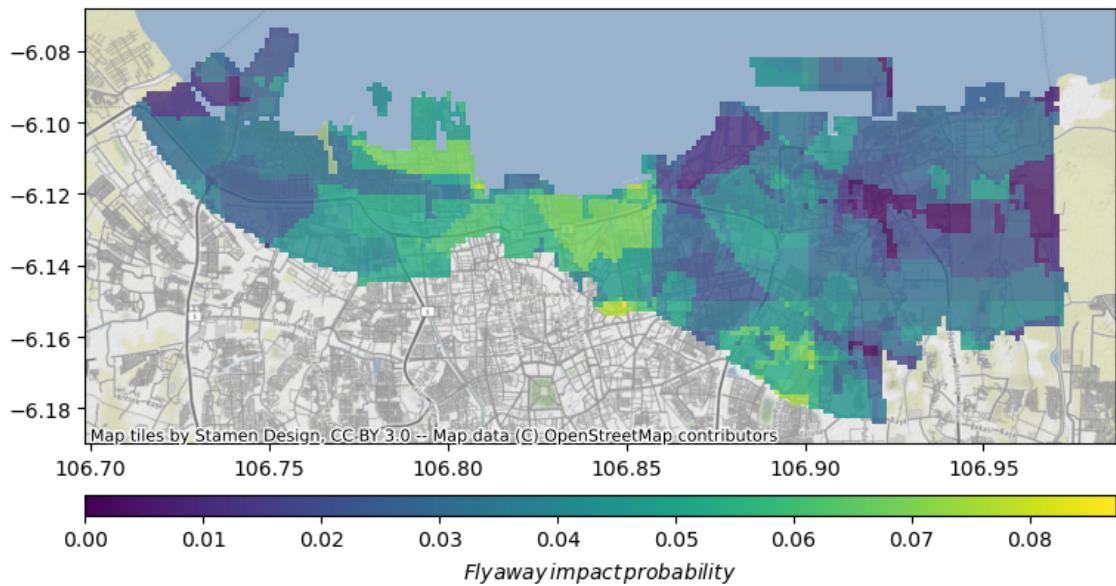


Figure 4.19: DJI Inspire 2 fly away PDF

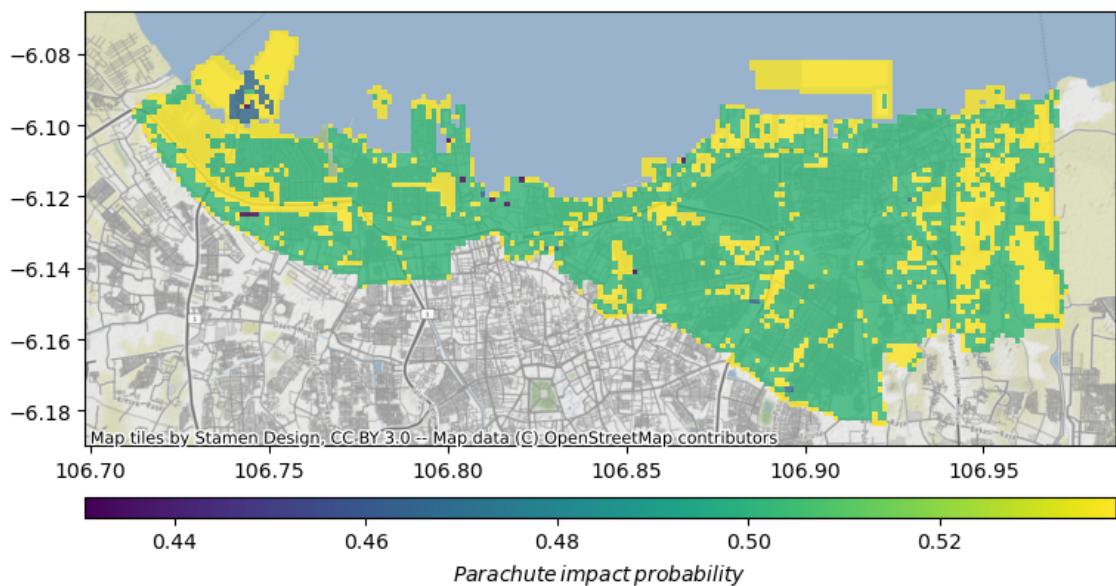


Figure 4.20: DJI Inspire 2 parachute PDF

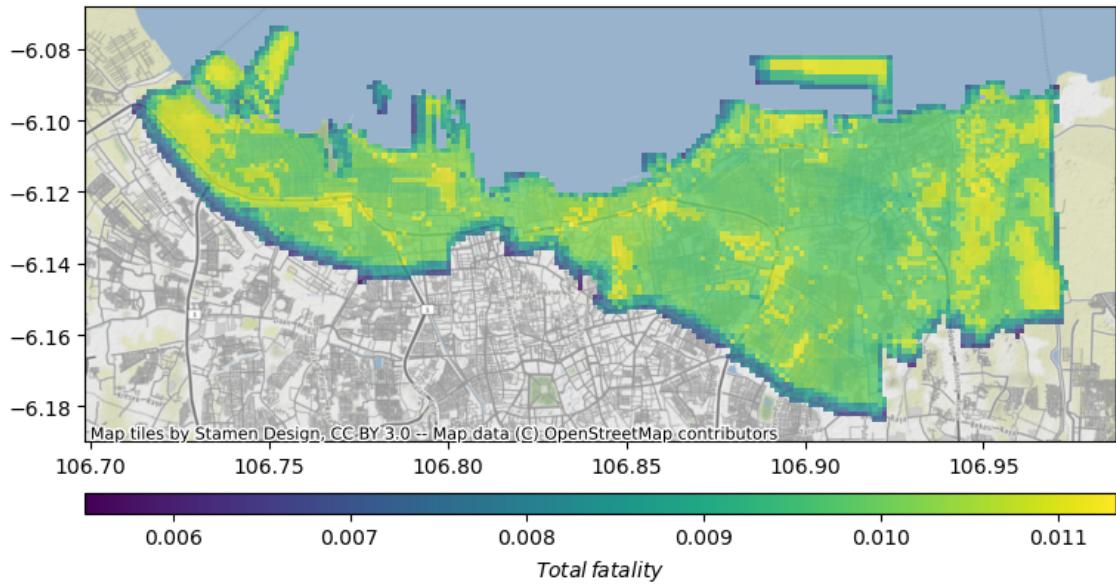


Figure 4.21: DJI Inspire 2 fatality risk PDF

4.4 Pathfinder result

4.4.1 Regular A*

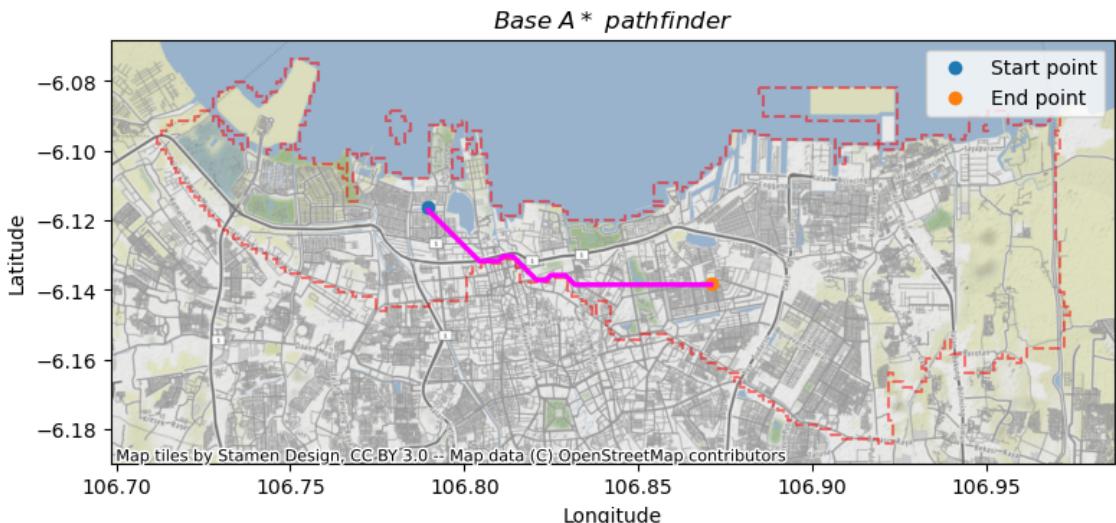


Figure 4.22: Regular A* flight path

4.4.2 Risk A*

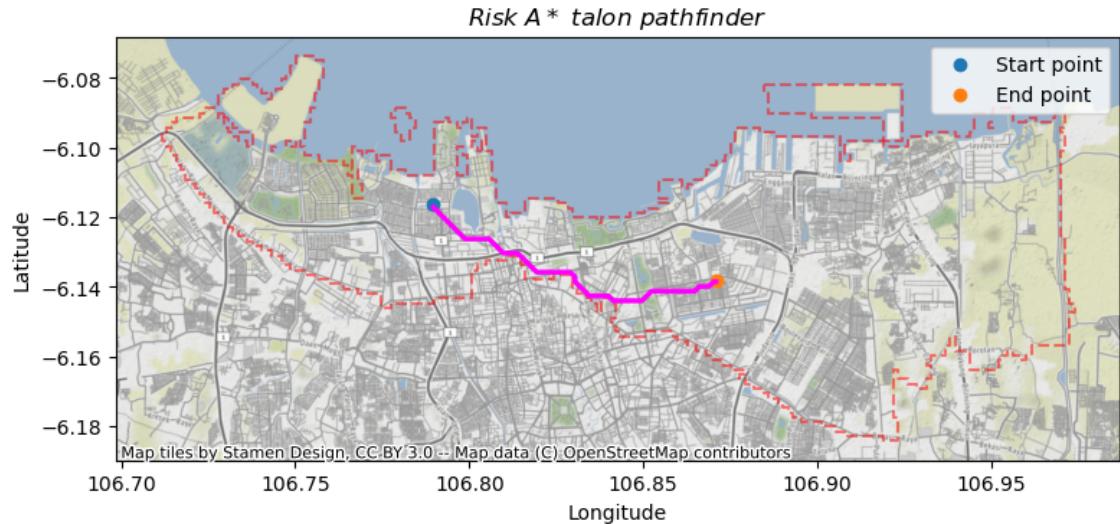


Figure 4.23: Talon risk A* flight path

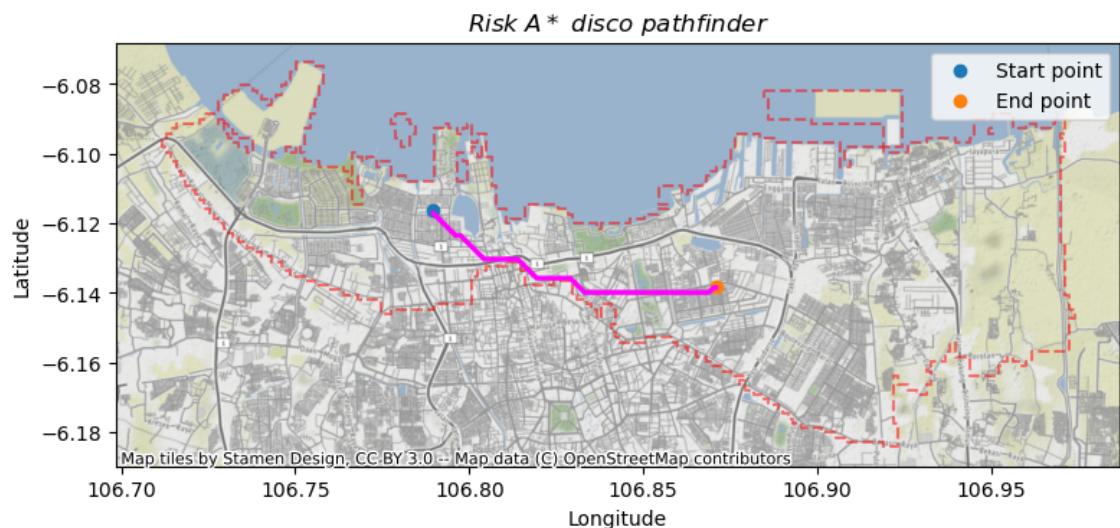


Figure 4.24: Disco risk A* flight path

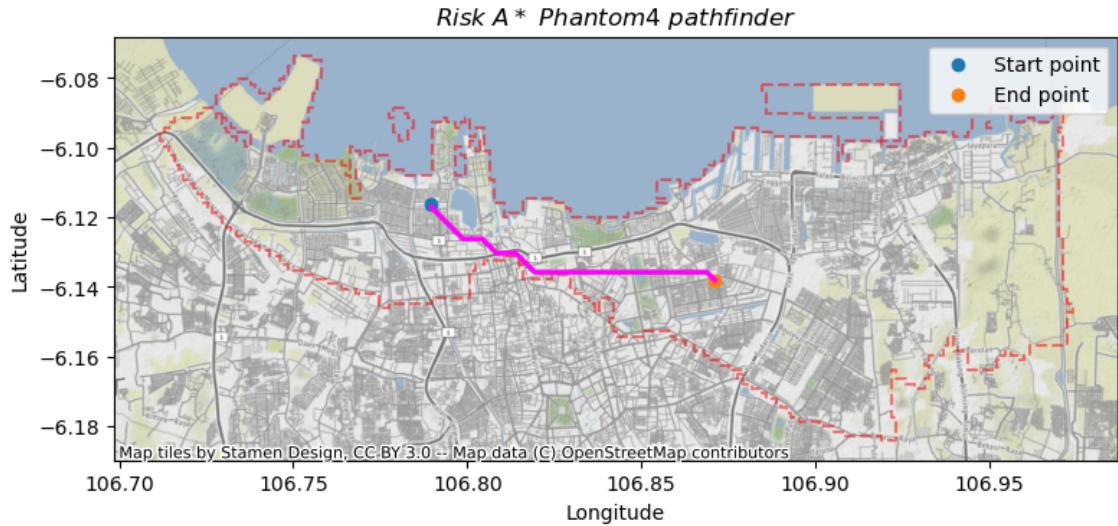


Figure 4.25: DJI Phantom 4 risk A* flight path

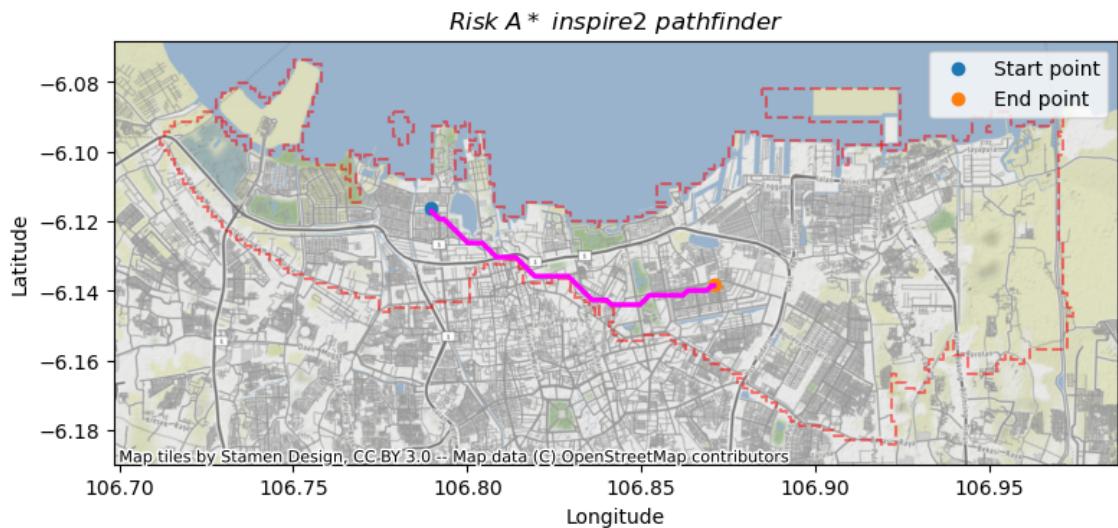


Figure 4.26: DJI Inspire 2 risk A* flight path

4.4.3 Risk result

From 4.4, It is not surprising that the traveled distance of Normal A* is the shortest of the rest of the risk A*. And also, it is interesting to see that the normal A* has the highest risk, although it is logical to think that normal A* inversely correlates to its risk A* counterpart. It is not guaranteed for a normal A* to have the highest risk score. However, normal A* will guarantee to have the shortest path length compared to risk A*.

To look at different properties, we can also measure the risk-to-traveled distance ratio, which can be thought of as how much the algorithm tried to avoid critical areas by extending the pathway to a much safer option. This example is clearly shown in DJI Phan-

tom 4, where DJI Phantom 4 shares almost the same risk as normal A* with a 0.02 difference, But takes a pathway that is 163 meters longer.

After looking at the graph, it is tempting to use Talon since it has the lowest traveled distance and also risk score. However, this resulted from the limitation in the way that the program is built, which of course, needs some improvement.

The problem lies in the Flyaway descent of this aircraft. This is because Talon has a long-range flight before crashing into the ground. Now the main problem is since we only use 1 cluster of geographic regions, North Jakarta, the region is smaller than the flyaway radius. This resulted in the outer cells of the region having a smaller risk compared to the center cells.

It is then clear the score represented by shorter-range UAVs such as Parrot Disco, DJI Phantom 4, and DJI Inspire 2 has a more defined score compared to the score produced by Talon.

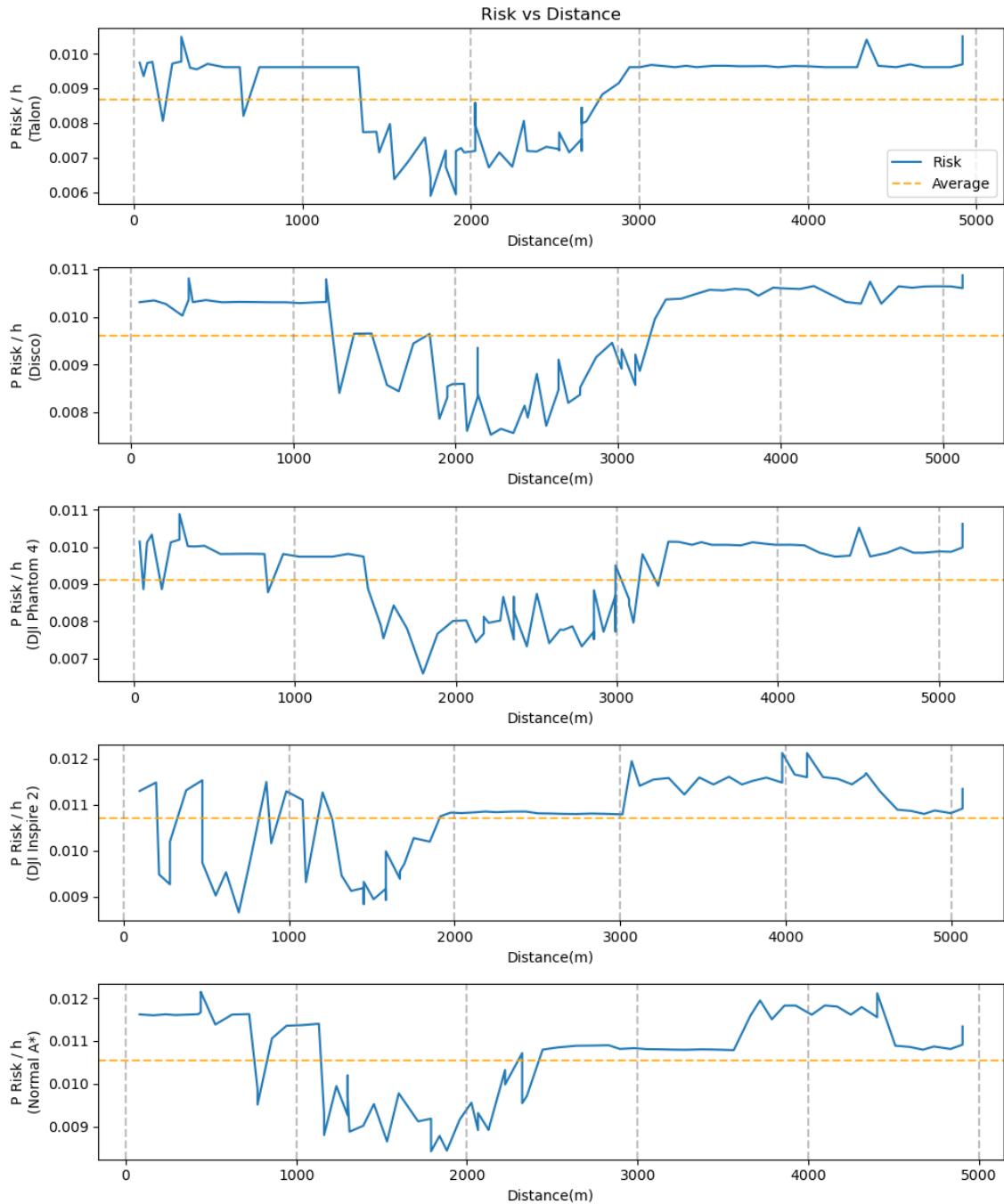


Figure 4.27: Aircraft risk score

Parameter	Talon	Disco	Phantom 4	Inspire 2
Total travelled distance	4917.651992	5120.845475	5066.796306	5147.428533
Total risk involved	0.727661	0.759534	0.834316	0.765401
Risk vs distance ratio	0.000148	0.000148	0.000165	0.000149

Table 4.2: Total pathway risk (%)

Drone	Stat	Ballistic	Glide	Flyaway	Final Risk
Talon	min	7.55e-02	3.14e-01	0.00e+00	5.36e-03
	max	5.38e-01	5.38e-01	7.76e-03	1.08e-02
	avg	4.74e-01	4.83e-01	8.47e-04	9.57e-03
Disco	min	5.63e-02	4.03e-01	5.38e-05	6.25e-03
	max	5.38e-01	5.38e-01	6.02e-02	1.13e-02
	avg	4.66e-01	5.06e-01	1.93e-02	9.91e-03
DJI Phantom 4	min	8.50e-02	3.99e-01	7.19e-03	6.85e-03
	max	5.39e-01	5.38e-01	1.79e-01	1.23e-02
	avg	4.87e-01	5.05e-01	9.19e-02	1.08e-02
DJI Inspire 2	min	6.03e-05	3.14e-01	0.00e+00	5.50e-03
	max	5.38e-01	5.38e-01	8.72e-02	1.13e-02
	avg	4.63e-01	4.83e-01	3.50e-02	9.81e-03

Table 4.3: Pathway risk statistic (%)

	Talon	Disco	Phantom 4	Inspire 2
Ballistic impact probability	3357.142553	3303.336594	3452.695201	3284.220663
Fly away impact probability	6.004905	136.521018	651.025645	248.005463
Glide impact probability	3420.454716	3584.184491	3576.983890	3420.454716
Parachute impact probability	3635.634721	3616.844608	3624.644006	3631.138685
Total fatality risk probability	67.836022	69.526808	76.807047	69.526808
Average areal total fatality risk	0.009572	0.009810	0.010838	0.009810

Table 4.4: Combine total map risk statistic (%)

CHAPTER 5

SUMMARY, CONCLUSION, AND RECOMMENDATION

5.1 Summary

In the summary ground, a risk map is a discrete PDF in the shape of a grid containing the possible fatality risk of each cell element. It is obtained by multiplying the possible outcome of multiple layers contributing to each risk. These layers are population density, no-fly zone layer, obstacle layer, shelter layer, and descent layer.

- The population layer is obtained by measuring the number of people per area of the cell.
- The no-fly zone layer represents military and aerodrome space and its critical infrastructure that the UAV must not fly into
- The obstacle layer represents a layer where the height of the geographic features becomes an obstacle depending on the flight level.
- The shelter factor represents the capability of the building to absorb the impact energy of a crashing UAV.
- The descent layer is a layer that describes the probability of the drone impacting the area. The descent layer is split into 3 parts that additively contribute to the descent layer. These parts are ballistic descent, glide descent, parachute, and flyaway.

After building the ground risk map. the next step would be finding the optimal Risk A* flight path where Risk A* is a modified A* pathfinding algorithm, which uses risk and distance to measure the most optimal path.

5.2 Conclusion

Using ground risk management to measure the total area fatality of impact risk resulted in an average of:

- Talon having 9.57×10^{-3} fatality per flight hour
- Parrot Disco having 9.91×10^{-3} fatality per flight hour
- Phantom 4 having 1.08×10^{-2} fatality per flight hour
- Inspire 2 having 9.81×10^{-3} fatality per flight hour

With a traveled distance of each aircraft being:

- Talon, with 4,918 m
- Parrot Disco, with 5,121 m
- Phantom 4, with 5,066 m
- Inspire 2, with 5,147 m
- Any Aircraft Without Risk involved, with 4,903 m

And average path risk of aircraft:

- Talon, with 1.48×10^{-4} fatality per flight hour
- Parrot Disco, with 1.48×10^{-4} fatality per flight hour
- Phantom 4, with 1.65×10^{-4} fatality per flight hour
- Inspire 2, with 1.49×10^{-4} fatality per flight hour
- Any Aircraft Without Risk involved, with 1.74×10^{-4} fatality per flight hour

5.3 Recommendation

5.3.1 For future research

5.3.2 For software design

BIBLIOGRAPHY

- Aalmoes, R., Cheung, Y., Sunil, E., Hoekstra, J., & Bussink, F. (2015). A conceptual third party risk model for personal and unmanned aerial vehicles. *2015 International Conference on Unmanned Aircraft Systems (ICUAS)*, 1301–1309. <https://doi.org/10.1109/ICUAS.2015.7152424>
- Ale, B., & Piers, M. (2000). The assessment and management of third party risk around a major airport. *Journal of Hazardous Materials*, 71(1), 1–16. [https://doi.org/10.1016/S0304-3894\(99\)00069-2](https://doi.org/10.1016/S0304-3894(99)00069-2)
- Ancel, E., Capristan, F. M., Foster, J. V., & Condotta, R. C. (2017). Real-time risk assessment framework for unmanned aircraft system (UAS) traffic management (UTM). *17th AIAA Aviation Technology, Integration, and Operations Conference*. <https://doi.org/10.2514/6.2017-3273>
- Ball, J. A., Knott, M., & Burke, D. D. (n.d.). CRASH LETHALITY MODEL.
- Barr, L. C., Newman, R., Ancel, E., Belcastro, C. M., Foster, J. V., Evans, J., & Klyde, D. H. (2017). Preliminary risk assessment for small unmanned aircraft systems. *17th AIAA Aviation Technology, Integration, and Operations Conference*. <https://doi.org/10.2514/6.2017-3272>
- Bleier, M., Settele, F., Krauss, M., Knoll, A., & Schilling, K. (2015). Risk assessment of flight paths for automatic emergency parachute deployment in UAVs. *IFAC-PapersOnLine*, 48(9), 180–185. <https://doi.org/10.1016/j.ifacol.2015.08.080>
- Breunig, J., Forman, J., Sayed, S., & Audenaerd, L. (n.d.). Modeling risk-based approach for small unmanned aircraft systems.
- Burke, D. A., Hall, C. E., & Cook, S. P. (2011). System-level airworthiness tool. *Journal of Aircraft*, 48(3), 777–785. <https://doi.org/10.2514/1.C031022>
- Clothier, R., Walker, R., Fulton, N., & Campbell, D. (2007). A CASUALTY RISK ANALYSIS FOR UNMANNED AERIAL SYSTEM (UAS) OPERATIONS OVER INHABITED AREAS.
- Cole, J. K., Young, L. W., & Jordan-Culler, T. (1997, April 1). *Hazards of falling debris to people, aircraft, and watercraft* (SAND-97-0805). Sandia National Lab. (SNL-NM), Albuquerque, NM (United States). <https://doi.org/10.2172/468556>
- Cour-Harbo, A. I. (2020). Ground impact probability distribution for small unmanned aircraft in ballistic descent. *2020 International Conference on Unmanned Aircraft Systems (ICUAS)*, 1442–1451. <https://doi.org/10.1109/ICUAS48674.2020.9213990>
- Dalamagkidis, K., Valavanis, K. P., & Piegl, L. A. (2008). Evaluating the risk of unmanned aircraft ground impacts. *2008 16th Mediterranean Conference on Control and Automation*, 709–716. <https://doi.org/10.1109/MED.2008.4602249>
- FAA's approaches to ground and NAS separation distances for commercial rocket launches [Aerospace sciences meetings] [Archive Location: world]. (n.d.). <https://doi.org/10.2514/6.2010-1540>
- Global human settlement - mission - european commission. (n.d.). Retrieved May 8, 2023, from <https://ghsl.jrc.ec.europa.eu/about.php>
- Harwick, M. J., Hall, J., Tatom, J. W., & Baker, R. G. (2007, February 2). *Approved methods and algorithms for DoD risk-based explosives siting*: Defense Technical Information Center. Fort Belvoir, VA. <https://doi.org/10.21236/ADA463571>

- Hazards to people and aircraft from flight test debris generated at high altitudes | aerospace sciences meetings.* (n.d.). Retrieved May 26, 2023, from <https://arc.aiaa.org/doi/abs/10.2514/6.1996-70>
- Kim, Y., & Bae, J. (2022). Risk-based UAV corridor capacity analysis above a populated area. *Drones*, 6(9), 221. <https://doi.org/10.3390/drones6090221>
- Klepeis, N. E., Nelson, W. C., Ott, W. R., Robinson, J. P., Tsang, A. M., Switzer, P., Behar, J. V., Hern, S. C., & Engelmann, W. H. (2001). The national human activity pattern survey (NHAPS): A resource for assessing exposure to environmental pollutants. *Journal of Exposure Science & Environmental Epidemiology*, 11(3), 231–252. <https://doi.org/10.1038/sj.jea.7500165>
- la Cour-Harbo, A. (2019). Quantifying risk of ground impact fatalities for small unmanned aircraft. *Journal of Intelligent & Robotic Systems*, 93(1), 367–384. <https://doi.org/10.1007/s10846-018-0853-1>
- Magister, T. (2010). The small unmanned aircraft blunt criterion based injury potential estimation. *Safety Science*, 48(10), 1313–1320. <https://doi.org/10.1016/j.ssci.2010.04.012>
- Melnyk, R., Schrage, D., Volovoi, V., & Jimenez, H. (2014). A third-party casualty risk model for unmanned aircraft system operations. *Reliability Engineering & System Safety*, 124, 105–116. <https://doi.org/10.1016/j.ress.2013.11.016>
- OpenStreetMap wiki.* (n.d.). Retrieved September 21, 2022, from https://wiki.openstreetmap.org/wiki/Main_Page
- Pat-Cornell, M. E. (n.d.). Uncertainties in risk analysis: Six levels of treatment.
- Primatesta, S., Guglieri, G., & Rizzo, A. (2019). A risk-aware path planning strategy for UAVs in urban environments. *Journal of Intelligent & Robotic Systems*, 95(2), 629–643. <https://doi.org/10.1007/s10846-018-0924-3>
- Primatesta, S., Rizzo, A., & la Cour-Harbo, A. (2020). Ground risk map for unmanned aircraft in urban environments. *Journal of Intelligent & Robotic Systems*, 97(3), 489–509. <https://doi.org/10.1007/s10846-019-01015-z>
- Shelley, A. (2016). A model of human harm from a falling unmanned aircraft: Implications for UAS regulation. *International Journal of Aviation, Aeronautics, and Aerospace*. <https://doi.org/10.15394/ijaaa.2016.1120>
- Small, U. A. S. (2009). Aviation rulemaking committee (ARC). *Comprehensive Set of Recommendations for sUAS Regulatory Development*, 1.
- Unmanned aircraft systems (UAS) registration task force (RTF) aviation rulemaking committee (ARC). (2015).
- Washington, A., Clothier, R. A., & Silva, J. (2017). A review of unmanned aircraft system ground risk models. *Progress in Aerospace Sciences*, 95, 24–44. <https://doi.org/10.1016/j.paerosci.2017.10.001>
- Weibel, R. E., & Hansman, R. J. (n.d.). Safety considerations for operation of different classes of UAVs in the NAS.
- Wu, P., & Clothier, R. A. (2012). The development of ground impact models for the analysis of the risks associated with unmanned aircraft operations over inhabited areas.