AIAA SCITECH 2024 Forum, 8-12 January 2024, Orlando, USA. Paper AIAA2024-0232
DOI:10.2514/6.2024-0232

Risk Assessment for sUAS in Urban Environments: A Comprehensive Analysis, Modelling and Validation for Safe Operations

Victor Celdran Martinez[1], Hyo-Sang Shin[2] and Antonios Tsourdos[3]
Cranfield University School of Aerospace, Transport and Manufacturing (SATM), Cranfield, Bedfordshire, MK43 0AL, United Kingdom

ABSTRACT

The rapidly growing number of applications for small Unmanned Aerial Systems (sUAS) in last-mile applications within metropolitan environments creates a complex airspace and ground safety scenario, where numerous risks must be considered to accomplish safe operations. The built-up and heterogeneously shaped geometry, together with the densely populated and transited nature of urban scenes, define a challenging scene for piloted and autonomous missions, where operators need to consider performance-based and third-party risks. In response to the increasing requirements inherent from urban scenarios, this paper proposes an integrated and comprehensive risk model for urban sUAS operations, composed by different risk layers designed for real-world scenarios, and validated through simulated drone flights and 3D risk-based navigation. By identifying the different risk requirements for sUAS operations, first party risks – including navigation performance, data link monitoring and collision avoidance – are computed within a photorealistic simulation environment for a discretized airspace representation. On top of this, trajectory based third party risks are modelled to identify potential routes subject to drone failure and consequent fatality and third party damage, as well as societal impact in terms of noise and privacy. Risk-based navigation techniques are implemented to validate the resulting model, including classical path planning and reinforcement learning. The results enable the perception of urban scenes associated risks through the lenses of risk modelling, providing a valuable methodology for sUAS urban operations and contributing to safer drone flights.

I. Introduction

The urban deployment of Unmanned Aerial Vehicles (UAVs) for applications like traffic monitoring, aerial photography, and delivery has observed an exponential growth, a trend anticipated to persist in metropolitan areas [1], [2]. The rapid adaptation of these technologies to society necessities defines a scenario in which the regulatory agencies are defining new frameworks for safe operations, especially in crowded urban environments. Inherent risks to sUAS operations within congested and complex layouts pose significant concerns in metropolitan UAV operations, including societal impact and accidents, leading to potential fatalities and property damages [3]. Operators must consider how the operations can be conducted in a safe manner, ensuring the drone will not be exposed to risks that could lead to aircraft failure and bystander impact or cause a major disruption [4], while at the same time think of the effect the flights might have on the population [5], [6]. On the one hand, these situations imply major costs for first parties – especially for costly professional aircrafts –, while on the other hand life-threatening conditions can take place involving third parties and result into serious injury or fatality. The understanding of the mission surroundings

[1] PhD Researcher, School of Aerospace Transport and Manufacturing
[2] Professor, School of Aerospace, Transport & Manufacturing
[3] Head of Center, Director of Research, School of Aerospace, Transport & Manufacturing, and AIAA Senior Member

needs to be considered an imperative condition for flight feasibility, quantifying the risks accurately rather than assessing qualitatively. Existing research has explored the challenges involved in sUAS operations in urban environments, covering from novel technology solutions for risk mitigation to risk assessment integrating population models in it. In urban settings, drone performance is challenged by RF and GNSS interference from buildings, affecting control and communication [7]. Furthermore, collisions with buildings or other aircraft pose property damage, injuries, and fatalities, particularly in densely populated regions [8]. Identifying these challenges, de-risking the urban airspace and operating accordingly are the baseline for a safer integration of sUAS within our cities. In terms of third party risk, studies have different risk integration models and navigation solutions. One study [9] suggested a model for a scene the size of Manhattan, where urban polygonal, GPS and census data where used as the base layer for A* and BIT* path planning algorithms. Another work [10] introduced a gravity model for population and traffic density for a 3D grid navigation problem, while similar approaches are seen for aerial vehicle delivery [11] and risk assessment for cost effective path planning [12]. While a wide range of metrics have been developed, there is a gap in accurate urban representation when applying the risks for sUAS operations.

Supporting the development of these risk assessment technologies a widespread practice is the implementation of urban 3D models, encompassing detailed representations of buildings, infrastructure, terrain, and other elements crucial for simulating drone flights. These models leverage Geographic Information Systems (GIS), Light Detection and Ranging (LiDAR), photogrammetry, and other mapping technologies to generate precise and high-fidelity virtual environments [13]. Advancements in gaming engines and simulation software have facilitated the development of interactive and customizable urban 3D models, allowing for the creation of diverse scenarios, as well as the training of deep learning models [14]. Furthermore, traditional simplified geometries are now challenged by the capabilities of drone urban mapping [15], data used to then populate urban models that enable further operations as some companies are already implementing [16]. The geometry digitalized data is translated into a set of associated risks, a conflicted airspace for which autonomous solutions enable scaling multiple platform operations to larger urban scenes. Methods for path planning and decision making in urban environments have been largely explored for terrestrial [17] [18] and threat-aware solutions [19], to airborne for multi UAV [20] and several path planning applications [12], [21]. Studies as the later provide cost-effective solutions for navigating the conflicted urban environments, modelling the risks involved with drone failure. Methods including genetic algorithms and Djikstra approaches are widely used to solve this kind of problem, together with classical A* path planning. Similar approaches use reinforcement learning [22], [23] to achieve risk aware navigation for complex resolution problem.

Throughout the numerous works in urban risks assessment for sUAS, the existing research provides a wide spectrum of detailed solutions for specific challenges, and while some authors explore multiple risks and present integrated risk assessment models, these rarely are scalable to the level of last-mile scenarios for a specific location in contrast to simplified or statistical buildings data. As a result, the operator – who needs to understand the risks involved with their operation – usually relies on qualitative assessment based on experience. In this paper the authors propose a comprehensive and integrated approach, where different techniques are explored and adapted to real-world 3D scenarios, validated through navigation algorithms and sample flight paths in order to shed light onto the feasibility of the results. The contributions of this article are summarized as follows.

1) This paper presents a complete tailored to 3D complex scenarios integrated risk assessment model, where the main mission-critical performance metrics provide an insight on how the urban geometry is perceived by the sUAS, on top of which third party trajectory-based risks are applied, providing a complete solution that accounts for mission performance concerns, fatality, damage and public disturbance scenarios.
2) This paper implements a photorealistic simulation environment, where the urban geometry mesh is loaded together with the relevant location specific urban data and serves as foundation for the drone flights modelling.
3) This paper validates the risk assessment through navigation techniques defining a risk-based navigation problem, solved using path planning and reinforcement learning; and a set of risk-representative flight paths to fully illustrate the risk model applicability. The performance geometry-based risk layers define the urban canyon navigation problem, while the third party trajectory dependent risk layers offer a more elaborate scope of the urban risks spectrum.

The remainder of the paper is structured in 5 sections. Section II identifies and characterizes the risks inherent to urban scenes for sUAS operations and introduces the model architecture. Section III presents in detail the proposed risk assessment methodology, reviewing and modelling the urban risks. Navigation algorithms for the model validation are explained in Section IV, followed by the results in Section V. Section VI concludes the paper.

II. Problem statement

Across the literature, different kinds of risk sources in urban environments have been identified and studied, ranging from GNSS resilience to property damage due to drone failure. Within the context of civil aviation, three different classes of risks are defined [11]: first party, referring to the air service provider; second party, concerning the part receiving the air transport service; and third party, involving the people exposed to the scope of the activities without any implication. Due to the unmanned nature of the sUAS operations, second party risks would refer to customer receiving the sUAS services rather than passengers and are therefore considered out of the scope for this risk model formulation.

First party risks comprise the set of circumstances and events the operator might face during the operation concerning the completion the mission successfully. Three main categories are identified as the fundamental factors to successfully deliver a flight for a sUAS:

1) data link robustness: communications robustness enables the aircraft's command and control, in addition to mission relevant data transmission,
2) navigation systems resilience, ensuring positional awareness in built up scenes,
3) and obstacle collision avoidance, found as the final layer of the platform's detect and avoid capabilities (DAA), crucial for the aircraft's integrity.

On the other hand, third party risks are characterized by how any event related to the mission might affect noninvolved parties. Previous works across the literature highlight the relevance of not only considering the impact of the sUAS failure, leading to property damage and life-threatening conditions, but also the impact of the flight and mission nature. The following risk layers are considered for the model accordingly:

1) fatality risks: people and vehicles on the ground are exposed to a drone impact in case of malfunction,
2) 3rd party damage: while some regions might be more congested than others in terms of obstacles, other might imply a higher damage risk due to the value or the incident impact might have,
3) and the societal impact, originated from the aircraft noise pollution and potential privacy concerns from camera recording.

Fig 1 shows a representative distribution of the risks sources is presented, including complex geometry impacting navigation and performance, in addition to ground traffic and population density.

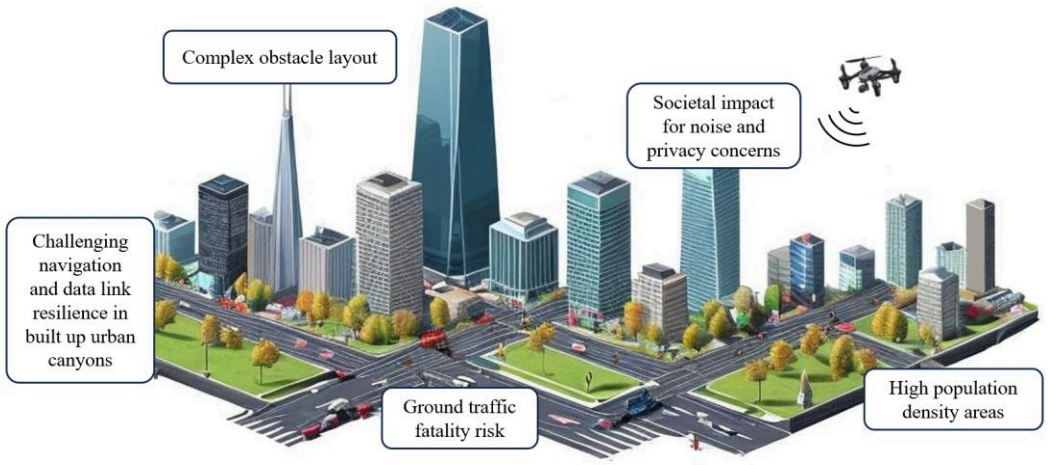
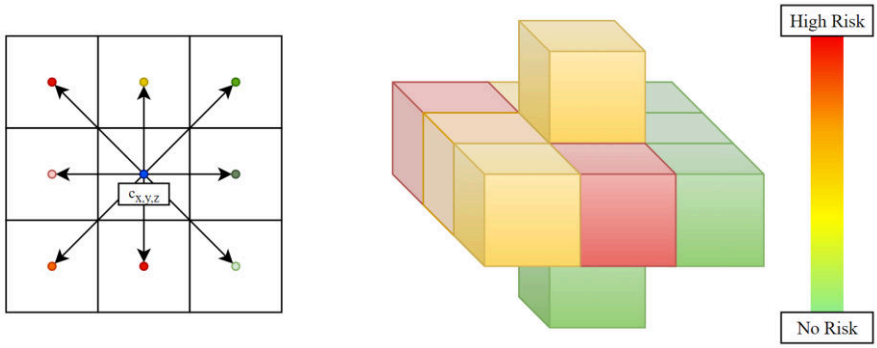


Fig 1. First and third party risks for sUAS operations in urban environments

The combined risk model is distributed across the scenario by means of a 3D division of the airspace into a voxel grid, defined by cubes with centroid c_{xyz} which are considered as the drone position when travelling through that voxel. The risk model is applied for each centroid defined across the airspace, integrating the first and third party risks for that coordinate and state. According the Joint Authorities for Rulemaking on Unmanned Systems (JARUS) [24], in relation to the Intrinsic UAS ground risk class classification, the shortest listed maximum UAS characteristic dimension is one meter, encompassing the vast majority of commercial drones. Therefore, the voxel cube dimension is established to one meter accordingly, effectively representing the risk-based urban environment.

Drone movement through the grid is modelled to travel from a given centroid to any of the 8 surrounding voxels in the same altitude level, applied to different flight levels rather than modelling a complete 3D movement (Fig 2). This motion model approximates UAVs actual operations motion between two consecutive waypoints, where the

aircraft can be oriented to any heading within the 360º and does not continuously altering altitude level simultaneously. Nevertheless, the model allows different flight level enabling a multi-altitude waypoint mission. During the pre-flight assessment, to travel from point A to point B where the latter point be is placed at a greater altitude, a waypoint following solution should first increase the altitude and then travel to B at z_B flight level, a safer option to avoid potential obstacles considering the B coordinates have been properly assessed in terms of collision avoidance.



(a) Horizontal grid motion (b) 3D representation of the voxel grid around a node

Fig 2. Drone risk map representation

The simulation environment is designed importing photorealistic urban scenes into the Unreal Engine 5 [25] game editor, together with the drone modelling library AirSim [26] for the drone's model and programmatic control. GNSS data is accessed from CelesTrak ephemeris data [27], while the ground risks related data is curated for each scene. In Fig 3 the first party risk are represented for a given drone location, in addition to the crash zone estimation trajectory for the calculation of the third party ground risks.

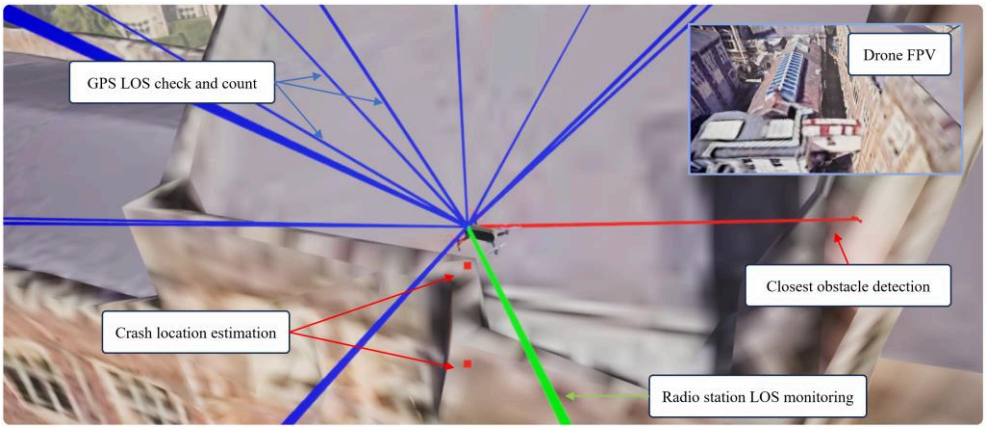


Fig 3. Risk model layers active in a 3D photorealistic environment

The simulations are set below the 400 feet ceiling, defined by the main aviation agencies [28]–[30], and the flight paths results are adapted to the maximum building ceiling. The data is processed within the airspace individual units and the integrated risk model provides a cost to travel through the given voxel, resulting into a risk-based navigation problem solved with reinforcement learning, previously validated through classical path planning, and applied to a set of trajectories.

III. Risk Model

This section presents the mathematical formulation for each risk layer and suggests an integrated cost function for the voxel grid centroids. As described in the previous section, two main categories of risks are identified and three different subcategories for each category are considered for the comprehensive risk model.

A. First party risks

Mission integrity related risks are subject to the urban geometry the mission takes place at, obstacles ranging from buildings to vegetation or lamp posts, among others. The mission critical nature of these risks implies that the cost

function not only needs to quantify the performance degradation suffered at a given coordinate, but also must provide no-go voxels that would result into a flight failure, including collision with an obstacle or completely losing navigation capabilities.

A.1. Navigation system layer

While there is considerable research on urban navigation systems that enable sUAS flights in GNSS-denied environments [31], in practical conditions most of the commercial options available for drone operators are strongly relying on satellite navigation together with technologies as IMU fusion, being dead reckoning the most widespread practice [32]. This paper proposes a geometrical calculation of navigation uncertainty cost, implementing satellite line of sight and computing the error of propagation form satellite geometry, known as Geometric Dilution of Precision (GDOP). Due to the complexity of accurately capturing GNSS phenomena in urban areas – as multipath and interference –, especially using 3D meshes, this paper considers out of scope the implementation of a complete navigation model, suggesting a navigation correction for the collision avoidance layer.

Satellite coordinates for a given time and date are extracted from CelesTrack and filtered for those located over the horizon at the simulation location coordinates. The resulting list of satellites is filtered by constellations, and assuming a multi-constellation capable sUAS, the calculations are performed using GPS, Galileo and GLONASS satellites. The final filtering stage performs LOS calculation within the simulation engine from each resulting satellite coordinates to each voxel centroid. The cost associated to crossing a given voxel centroid is calculated from the navigation uncertainty based on the GDOP value. For a given set of satellites position (x_i, y_i, z_i) and a receiver position (x_r, y_r, z_r), the range can be obtained as:

$$r_i = \sqrt{(x_i - x_r)^2 + (y_i - y_r)^2 + (z_i - z_r)^2} \tag{1}$$

The design matrix A for each satellite is constructed from the partial derivatives of the range equations with respect to the receiver's position, used to obtain the covariance matrix Q:

$$A = \left[\left[\frac{(x_i - x_r)}{r_i}, \frac{(y_i - y_r)}{r_i}, \frac{(z_i - z_r)}{r_i}, -1\right]\right] \qquad Q = (A^T \cdot A)^{-1} \tag{2}$$

From the covariance matrix Q, the diagonal elements can be extracted,

$$HDOP = \sqrt{Q_{[1,1]} + Q_{[2,2]}} \qquad VDOP = \sqrt{Q_{[3,3]}} \tag{3}$$

$$PDOP = \sqrt{Q_{[1,1]} + Q_{[2,2]} + Q_{[3,3]}} \quad TDOP = \sqrt{Q_{[4,4]}} \tag{4}$$

and the final value of GDOP obtained as

$$GDOP = \sqrt{PDOP^2 + TDOP^2} \tag{5}$$

The cost associated to crossing the voxel due to GNSS uncertainty is:

$$c_{GNSS} = 1 - \frac{GDOP_i - GDOP_{min}}{GDOD_{max} - GDOP_{min}} \qquad c_{GNSS} \in [0,1] \tag{6}$$

The classification of DOP values are presented in Table 1 [33], which for close-to-obstacles operations as in urban scenes a standard convention for UAVs is aiming for a value of GDOP lower than 6, resulting in $GDOP_{max}$ of 6 for the cost calculation, and a $GDOP_{min}$ of 1. This risk is considered as a no-flight condition for cost scores of 1, as navigation resilience cannot be ensured. Furthermore, in conditions of not enough satellites, 5 for the case of multi-constellation GNSS receivers, the cost score is set to maximal as well.

Class number	GDOP value	Ratings
Class 1	1	Ideal
Class 2	2-3	Excellent
Class 3	4-6	Good
Class 4	7-8	Moderate
Class 5	9-20	Fair
Class 6	21-50	Poor

Table 1. GDOP values classification [ref]

A.2. Data link layer

A continuous data link between the sUAS and the operator is required for safe operations conduction, as while the technologies move forward autonomy the regulations require from the operator surveys for frequency coverage throughout the potential operating area [34]. Several studies have been performed to model the RF coverage for drone operations through metropolitan scenes [35], however, most of the models consider simplified urban geometry in order to incorporate the complex phenomena RF experiments due to the urban geometry, including reflection or diffraction, among others.

This study does not consider the development of a complete drone data link coverage model, and rather suggests an applicable tool for complex and real urban geometry. Based on an average drone operation, where either the pilot or the ground station usually stands next to the drone take off site, the antenna is modelled at an advantageous elevated location – selected during the site assessment – and the associated risk assessment cost is conducted checking for the line of sight between the antenna and each voxel location. Given the complexity of modelling propagation in complex 3D meshes without dedicated software and hardware, rather than defining a more exhaustive risk map as the navigation layer, this simplified data link layer provides potential coverage loss regions. As a result, the cost is modelled as follows:

$$c_{data} = \begin{cases} 1, & if\ \exists\ LOS \\ 0, & otherwise \end{cases} \tag{7}$$

A.3. Collision avoidance layer

The final first party risk considers the urban geometry to, on the one hand, define no-flight regions due to obstacle presence in the voxel, and on the other hand provide a risk profile close to all the obstacles, descending the cost when increasing the distance to an obstacle. In order to achieve that, the model takes advantage of Unreal Engine's capabilities to determine the voxel occupation, creating a coarse point cloud from any given 3D mesh map which turns into the base layer for the collision avoidance cost.

From the occupied voxel coordinates, a breath first search algorithm is applied to identify those voxels found within a distance R_{min} from the occupied voxels, creating a database updated whenever for a common close-to-obstacle empty voxel a smaller distance to an obstacle than the initial is found. As a result, all empty voxels are assigned a distance to the closest obstacle, except for those found at a greater distance than R_{min} to any obstacle on the map.

The cost for the collision avoidance layer is modelled as an inverse exponential function, where the maximum cost value is given at the obstacle and the cost decreases with the distance according to the following expression:

$$c_{CA} = \begin{cases} 1, & d_{obs} = 0 \\ e^{-m_{GNSS}\frac{d_{obs}}{R_{min}}}, & 0 \le d_{obs} \le R_{min} \\ 0, & d_{obs} > R_{min} \end{cases}, \quad m_{GNSS} = m_0(1 - c_{GNSS}) \tag{8}$$

where d_{obs} is the distance to the closest obstacle and m_{GNSS} is the slope of the cost function dependent of the value of the GNSS layer, based on a minimal slope for no GNSS cost $m_0 = e$ – meaning the greatest risk reduction rate possible. Collision avoidance is strongly relying on the navigation system performance, and therefore this risk layer considers a higher risk of crashing into an obstacle for a given distance when the GNSS receiver performance is degraded.

In the same way as the GNSS layer, c_{CA} also define no-flight zones, overriding the integrated risk model cost function for a given voxel when maximal risk takes place, meaning an obstacle is present.

B. Third party risks

In contrast to the first party risk layers, third party risks are highly conditioned by the drone dynamic states. While some studies model the ground risks considering the cost for as given voxel is determined by the ground data right below it, as population density, this paper suggests an alternative approach and integrates the speed vector to predict a crash location. Due to the reduced dimensions of the airspace units within the voxel grid earlier introduced, in contrast to larger grid unit studies, the drone descent trajectory cannot be assumed to fall within the same Z coordinate the drone suffers the failure at. Therefore, a drag conditioned drop is modelled from the failure voxel using the initial velocity. The first occupied voxel the trajectory crosses, either horizontal – the ground, a rooftop – or vertical, as a building wall, determines the drone impact energy. As a result, third party risks are modelled as trajectory-based risks, being applied to each attempted drone velocity vector - for the navigation algorithm – set of crossed voxels from start to goal during flight. Failure based risk layers are a function of the drone impact energy, given by the drone descent velocity. The acceleration the drone suffers while descending, affected by the aerodynamic drag with no wind conditions, can be modelled as

$$a_{aero} = \frac{F_{aero}}{m} = \frac{c_d \rho_{air} S_d v_{air}^2}{2m_d}, \qquad a_y = \frac{F_{grav} - F_{aero_y}}{m} = g - a_{aero_y}, \qquad a_x = -\frac{F_{aero_x}}{m} = -a_{aero_x} \tag{9}$$

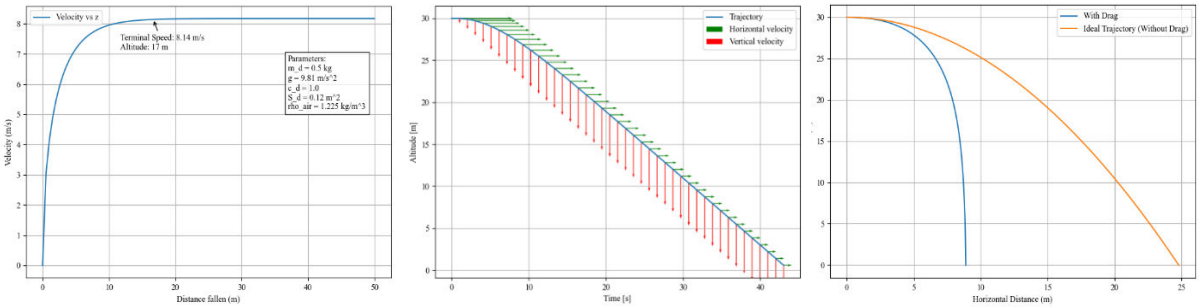
where g is the gravitational acceleration 9.81m/s^2, c_d is the drone drag coefficient, S_d and m_d are the drone's cross-sectional surface and mass, ρ_{air} is the air density, and v_{air} is the air speed of the drone. Assuming the drone reaches a stable, constant velocity after falling a certain distance, the drone vertical terminal velocity can be expressed as

$$v_y(z_d) = \sqrt{\frac{2m_d g}{c_d S_d \rho_{air}} \left(1 - e^{-\frac{c_d S_d \rho_{air} z_d}{m_d}}\right)} \tag{10}$$

where z_d is the vertical coordinate the drone is at from the ground.

Given the wide range of sUAS configurations and geometries, a representative drone is modelled at c_d of 1.0 [36] according to a Parrot Anafi USA. A vertical terminal speed of 8.14 m/s is obtained, which for a drone mass of 0.5kg results into an impact energy of 16.75J, less than half of the equivalent to being hit by a tennis ball serving, for example. Kinetic energy greatly increases with the drone weight, as it also increasing the terminal speed value, obtaining four time as much energy for double the mass.

In Fig 4 (a) the graph shows how for different drop altitudes the drone impact vertical velocity tends to a vertical terminal speed – taken at 99% of the curve – with the increase of the altitude for a representative drone. On the other hand, graph (b) displays for the same set of parameters, both velocity components throughout the trajectory. Note that while the horizontal speed continuously decreases from an initial 10m/s value, the vertical component achieves a steady value of 8.17m/s, and therefore a constant rate of altitude value decrease. The horizontal speed considerably for a drop altitude of 30m, while for earlier altitude crashes, as a 10m descent before finding an obstacle, the total speed has a larger influence from the horizontal component. While the speed might remain a similar value without considering the effect of the drag in the horizontal velocity component, the resulting crash location considerably varies – reaching nearly a third of the distance for an ideal parabolic trajectory, visible in (c), for a cruise speed of 10m/s.



(a) Terminal v_y vs the drop height (b) Drone Altitude vs Velocity with drag (c) Drone distance with and without drag

Fig 4. Drone speed against drop altitude and terminal speed (left), velocity components (right) and trajectory (bottom)

B.1. Fatality risk layer

The fatality risk layer aims to model the effect of a potential drone crash over a person or vehicle which could lead to a fatality. Regarding the risk related to bystanders, a gravity model is implemented to distribute the population density around key more frequented locations, as a train station, a coffee shop or a museum, for an average density of 0.011/m^2 found in Central London [37]. The resulting population distribution has an averaged density equal to the total map density, decreasing the density from further point to the key locations and adding it to those closer.

Road traffic is modelled according to the size of the roads as a scaling factor, assuming for a given average representative road with a set width and traffic, a larger road would not only provide more cells of road traffic, but also the density of these cells is greater as it would attract a higher affluence of cars, especially considering how modern cities aim to reroute inter-city traffic to larger roads; and vice versa for smaller roads. Regarding the cost functions, the model adapts recent third party modelling research [10] to the trajectory based risk modelling, defining the bystanders fatality risk as

$$c_{fat_p} = p_{fail} n_{hit}^p r_f^p \qquad n_{hit}^p = S_{hit}\rho_p \qquad \rho_p = f(x_c, y_c, z_c, \rho_p^{avg}) \tag{11}$$

where p_{fail} describes the probability of aircraft failure, n_{hit}^p is the number of people hit, defined by the drone surface and the voxel corresponding population density ρ_p, and r_f^p is the fatality rate associated with a drone impact. The value for r_f^p is estimated based on drone human crash testing research [38], for which the probability of a serious injury in terms of skull fracture is of 5.09% for a drone of mass of 680g at a speed of 13m/s – greater values compared to the used drone mass and terminal velocity, and therefore providing a safety margin in obtaining a higher impact risk.

Similarly, for a vehicle impact the cost can be described as

$$c_{fat_v} = p_{fail} n_{hit}^v r_f^v \qquad n_{hit}^v = S_{hit}\rho_v \qquad \rho_v = f(x_c, y_c, z_c, \rho_v^{avg}) \tag{12}$$

where r_f^v in this case refers to the fatality rate associated to a traffic accident happening and resulting into fatality and ρ_v is the calculated density of vehicles for that map region based on the city averaged cars density ρ_v^{avg}. A drone impact into a windshield at terminal speed is not likely to result into the glass shattering, and therefore the source of accident would be more related the distraction the impact would generate. Therefore, r_f^v is computed as the product of the fatality probability for a crash incident and the probability of the car incident occurring due to the drone impact.

$$r_f^v = r_{f_{acc}}^v p_{acc}^v \tag{13}$$

The values for the cost expression are presented in Table 2, and the resultant risk cost value associated with the fatality risk is modelled as the addition of both risks.

$$c_{fat} = c_{fat_p} + c_{fat_v} \tag{14}$$

Risk factor	p_{fail}	S_{hit}	r_f^p	$r_{f_{acc}}^v$	p_{acc}^v	ρ_p^{avg}	ρ_v^{avg}
Value	$6.04 \cdot 10^{-5}$	0.12 m^2	0.0509	0.27	0.3	0.011/m^2	0.008/m^2

Table 2. Fatality risk cost function factors

B.2. Property damage layer

To accurately measure the property damage risk for each different real-world scene, considering the actual buildings and other property, the collided obstacle would need to be identified. However, rather than suggesting an obstacle recognition layer, this paper implements an approximation based on the following assumption: the drone is very likely hit a building or urban element if the crash coordinates are greater than a minimum height over the ground, under which it is considered to have crashed into the ground. The complexity of the drone dynamics after the first crash are considered out of scope, exclusively considering the first drone collision location.

The consequence of an impact on third party property is modelled according to a prone-to-be-damaged factor, according to the obstacle hit. Three main categories are defined, being the first those regions with little to no risk – as a green area or park with trees – identified and encapsuled within a polygon in the map; other flat open ground level

surfaces, as roads or parking sites – identified as obstacles with altitude level under 2 meters; and other than trees vertical elements, mainly representing buildings, corresponding to crashes occurred over 2 meters above the ground.

The impact energy is also considered from the velocity model earlier presented, using the drag-influenced velocity and normalized by the maximal kinetic energy, taken at the maximal speed point throughout the trajectory – terminal speed or initial speed if greater.

$$c_{prop} = p_{fail} r_{KE} k_{obs}\,, \quad r_{KE} = \frac{E_{k_{impact}}}{E_{k_{max}}} = \frac{v_{impact}^2}{v_{max}^2}\,, \quad k_{obs} = \begin{cases} 0.2, \ if\ green\ area \\ 0.6, \ if\ open\ urban\ area \\ 1, \ if\ urban\ obstacle \end{cases} \tag{15}$$

where r_{KE} is the ration with the maximal possible kinetic energy and k_{obs} is the obstacle damage factor.

B.3. Societal impact layer

The final third party layer considers the impact a drone operation might have over the population when operations take place with no incidents. In this category, the main challenges are noise pollution and privacy concerns from video cameras.

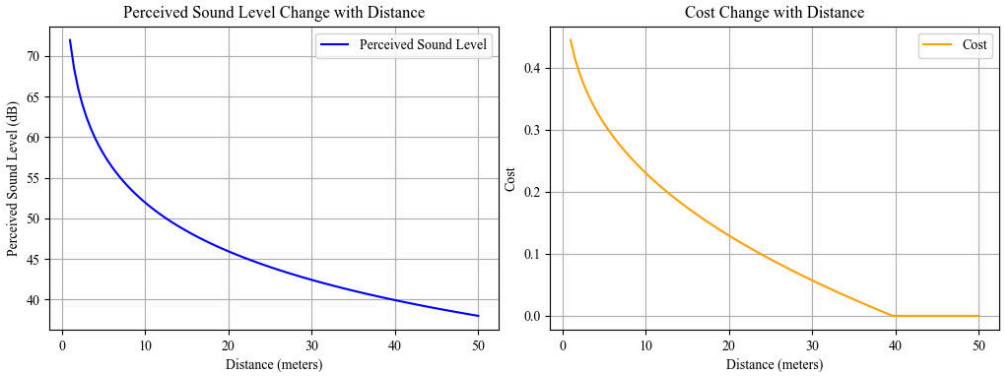
Noise pollution is modelled using a spherical sound propagation model from the drone center (reference), corrected for a given drone speed against a given observer speed using Doppler effect correction.

$$L_{propagated}(r) = L_{emitted} - 20log\left(\frac{r}{r_{ref}}\right) \qquad f' = f\left(1 + \frac{(v_o - v_d)}{v_s}\right) \tag{16}$$

$$L_{perceived}(r, v_s, v_o) = L_{emitted} - 20log\left(\frac{r}{r_{ref}}\right) + \Delta L_{Doppler} \qquad \Delta L_{Doppler} = 20\ log\left(\frac{f'}{f}\right) \tag{17}$$

where $L_{emitted}$ and r_{ref} are 40dB and 40m respectively [10], r is the distance to the closest population density positive voxel, f' corresponds to the Doppler effect frequency adjustment, f the drone emitting frequency set at 500Hz according to research frequencies under this threshold are strongly affected by background noise [39]; v_s, v_o and v_d the speed of sound, observer and drone respectively; and $L_{perceived}$ is the sound level perceived by an observer for a given observer and drone speeds. The cost function for the sound pollution impact results as follows, visible in Fig 5:

$$c_{sound} = \frac{L_{perceived} - L_{emitted}}{L_{perceived}} \tag{18}$$



(a) Perceived sound level vs distance to the source (b) Sound pollution cost evolution with distance

Fig 5. Noise level and cost change with distance for a drone and observer speeds of 5 and 2 m/s respectively

In terms of privacy, modelling the impact of a video camera recording is a complex task. The model from [40] is adapted, where the authors approximate a regression formula based on distance with and obtained 0.765 accuracy over experimental results:

$$z = 33.65 - 1.28x - 0.63y + 0.02x^2 + 0.007y^2 \tag{19}$$

where z is the number of pixels, and x and y are the horizontal and vertical distance to the object respectively. Another study [11] suggests the baseline for recognition is established at 10 horizontal pixels implying a 70% rate of recognition. For example, for a x and y values of 10 meters, the number of pixels obtained is ≈17, making the potential person in the video recognizable. As a result, the privacy cost is defined as:

$$c_{priv} = \begin{cases} 1, & if\ z > 10 \\ 0, & otherwise \end{cases} \tag{20}$$

Sound pollution and privacy costs are integrated within the societal impact layer using weighting factors k_{sound} and k_{priv} set to 0.5 each, considering both factors equally relevant in terms of public impact.

$$c_{soc} = k_{sound}c_{sound} + k_{priv}c_{priv} \tag{21}$$

B.4. Third party risks cost function

The three different layers are combined as a weighted function where each individual cost has already been normalized, being the corresponding weights $k_{fat} = 0.6,\ k_{prop} = 0.2,\ k_{soc} = 0.2$.

$$c_{3^{rd}_{party}} = k_{fat}c_{fat} + k_{prop}c_{prop} + k_{soc}c_{soc} \tag{22}$$

C. Integrated risk cost function

The static voxel grid cost values, defined by the first party risks, define high risk and no-flight regions unlike the third party risks. As a result, the voxel grid's first party associated cost is defined as follows:

$$c_{1^{st}_{party}}(x,y,z) = \begin{cases} 1, & if\ c_{GNSS} = 1\ or\ c_{CA} = 1 \\ \Sigma_1^3 k_i c_i, & otherwise \end{cases} \tag{23}$$

where the individual weighting factors are set as

$$k_{GNSS} = 0.4 \quad k_{data} = 0.2 \quad k_{CA} = 0.4 \tag{24}$$

A stronger weight is assigned to the metrics mostly responsible for the aircraft's integrity loss. The reason for this weights distribution is since modern sUAS are equipped safety features including return to home for data link loss scenarios, during which the navigation systems is required to be fully functional.

Third party risks, modelled as trajectory risks, are applied to each potential voxel the aircraft would need cross with the corresponding speed vector. The weight for the fatality risk is considered to be the most important factor in determining third party risks as it has the largest impact. Property damage and societal impact are given lower weights and considered at the same level of impact, as while property damage might imply a higher cost for the operation, the societal impact is perceived more frequently by the concerning third parties. Furthermore, given the different order of dimensions from the different risk layers, a scaling factor ω_i is introduced, normalizing the risks against the maximum possible value for each risk layer.

$$c_{3^{st}_{party}}(x,y,z,v) = \Sigma_1^3 k_i \omega_i c_i\ , \qquad k_{fat} = 0.6 \quad k_{prop} = 0.2 \quad k_{soc} = 0.2 \tag{25}$$

The integrated risk cost for a given voxel centroid c_{xyz} is therefore defined as:

$$c_i(x,y,z,v) = \begin{cases} 1, & if\ c_{1^{st}_{party}}(x,y,z) = 1 \\ k_{1^{st}_{party}}c_{1^{st}_{party}} + k_{3^{rd}_{party}}c_{3^{st}_{party}}, & otherwise \end{cases} \tag{26}$$

In order to define the weights for each main risk category, the scenario where the operations take place need to be considered. A very complex urban layout requires a thorough obstacles assessment as well as considering third party risks whenever there are elevated population and traffic densities. This scenario would require a tradeoff between first and third party risk, properly defining the operation requirements and how the modelled risk would affect it, assuming a level of acceptable risk level. On the other hand, for operations in segregated air and ground airspace, including complex geometry – for example, a power plant inspection –, first party risks would take a greater weight, while for a very crowded event operation – as an open-air concert –, third party risk are considerably more relevant as drone

performance is not expected to be considerably degraded, while population density greatly increases. For this study, both risks are set at an equal weight of 0.5 each for a representative scene where both risks are considered equally relevant.

IV. 3D risk-based navigation validation

The navigation solution is formulated as minimum cost navigation problem. This section presents firstly a validation of the first party risks – static grid risks – using A* path planning and Double Q-learning Reinforcement Learning (DQN-RL), for which the aim is to compare both solutions with a set of metrics in order validate the effectiveness of the first party risks by two different approaches. The second part of this section extends the solution for the dynamic trajectory model, for which the DQN-RL is used to handle the complex solution for the resulting large dataset and the dynamic trajectory associated risks navigation. The integrated dynamic trajectory risk model is tested for a set of missions and the feasibility of these is analyzed.

A. First party risks navigation validation

The three defined first party risks provide a voxel grid where each voxel centroid has an associated $c_{1^{st}party}$ cost value for a $(N_x x N_y x N_z)$ grid where each risk ranges from 0 to 1. The actions are defined as the 8 potential voxel movements introduced earlier, enabling all 8-direction possibilities on the horizontal plane.

The A* risk-based path planning algorithm cost represents the cost $c(i,j,k)$ associated with moving from centroid (i,j,k) to its neighboring centroids. First the start node, open list and close list are initialized. The heuristic $h(n)$ function estimates the cost from a node (i, j, k) to the goal using Manhattan distance calculation, corresponding to the accumulated risk achieving the goal would imply, while the cost $c(n)$ function represents the actual cost to reach each node from the start node, also corresponding to the accumulated cost.

For each iteration, the start node is added to the open list with estimated overall cost $C(i,j,k) = g(i,j,k) + j(i,j,k)$, and the algorithm iterates selecting the node with lowest $C(i,j,k)$, which becomes the current node and it is passed onto the closed list. While the goal is not achieved, nodes are expanded, either adding or updating values for each node found in the open list. If the successor is in the open list with higher $C(i,j,k)$.Once the goal is achieved, the path is reconstructed tracing back from the goal node thorough parent nodes stored in the closed list. As a result, the algorithm combines Djkstra's uniform cost search accuracy with heuristics to find the optimal path, enabling minimizing both cost and risk exposure.

While providing the optimal solution, graph-based methods can be computational expensive, especially for large and complex scenarios as the one this paper defines. Within this gap reinforcement learning techniques offer a cost-effective solution, which while it might provide an approximate result, the applicability to repetitive flight assessments and numerous flights defines a set of requirements more suitable for these techniques.

Double Q-Learning is used for this problem, and algorithm is as follows:

1) Initialization

a. Q-value initialization: initialize Q-value functions Q_1, Q_2 for all state-action pairs in the grid.

$$Q_1(s,a), Q_2(s,a) \quad for\ all\ s \in States, a \in Actions \tag{27}$$

b. Parameters initialization: set the exploration rate ϵ and discount factor γ for learning.

2) Reward function

a. The reward function encourages the agent to reduce the distance to the goal while being penalized for crossing high risk voxels:

$$r = f(s,a,s') = r_{dist}(x,y,z) - k_c c_{risk}(x,y,z) \tag{28}$$

where r_{dist} is the reward based on the Euclidian distance to the goal, k_c is a scaling factor for the risk and $c_{risk}(x,y,z)$ is the cost associated with travelling to the next voxel.

3) Training Phase

a. Action selection and exploration:

$$action = \begin{cases} random\_action(), & with\ probability\ \epsilon \\ \arg\max_a (Q_1(s,a) + Q_2(s,a)), & with\ probability\ 1-\epsilon \end{cases} \tag{29}$$

b. Perform action and update Q-values:

$$Q_1(s,a) \leftarrow Q_1(s,a) + \alpha \cdot [r + \gamma \cdot Q_2(s', \arg\max_a(Q_1(s',a))) - Q_1(s,a)]$$
$$Q_2(s,a) \leftarrow Q_1(s,a) + \alpha \cdot [r + \gamma \cdot Q_1(s', \arg\max_a(Q_1(s',a))) - Q_2(s,a)] \quad (30)$$

where α is the learning rate, s and s' represent states, and argmax_a selects the action maximizing the Q-value.

c. Exploration decay: reduce the exploration rate ϵ over iterations to shift from exploration to exploitation.
d. Training is performed for N_t training steps set according to map size requirements.

4) Navigation

a. The obtained Q-values are applied to find the optimal path from point A to point B while avoiding states above an established risk threshold.
b. Select movement based on highest Q-values minimizing path risk exposure.

B. Integrated first and third party risks navigation

The algorithm for the integrated first and third party risk model uses the same structure, extending the reward function to the velocity dependent third party risks, obtained as follows:

1) Descent trajectory: For a given current voxel (x, y, z) and velocity vector (v) the drop trajectory is modelled and the first collision voxel identified, obtaining (x_c, y_c, z_c).

2) Accident-based risk layers calculations

a. Fatality risk is obtained as $c_{fat} = c_{fat_p} + c_{fat_v}$ from the ground data for the x_c, y_c
b. Property damage risk applies the drone speed to obtain the kinetic energy ration, applied to the ground data at x_c, y_c.

3) Societal risk calculations: for each (x, y, z) travelled voxel the noise and privacy impact is calculated.

4) Final third party risk: the resultant third party risk cost is computed from the layers and weighting factors.

5) Integrated cost: the final cost to travel to the next state is computed using (26).

V. Simulation results

The simulation environment for the study is set in London in the centric area of Marylebone, incorporating a wide range of urban elements (Fig 6), for which the drone transmitter antenna is placed onto a rooftop strategically chosen for operating in the area, centered at the coordinates (51.515995, -0.156999). The resulting 3D first party risk map can be seen in Fig 7 for two different risk levels representation. Note how the urban canyons are clearly defined for an assumed risk over the 50%, where the red voxels indicate a very high or maximal level of risk, defined by the collision and GNSS layers; however, when considering all voxels with an associated risk, the safest option is flight over buildings level, as expected.

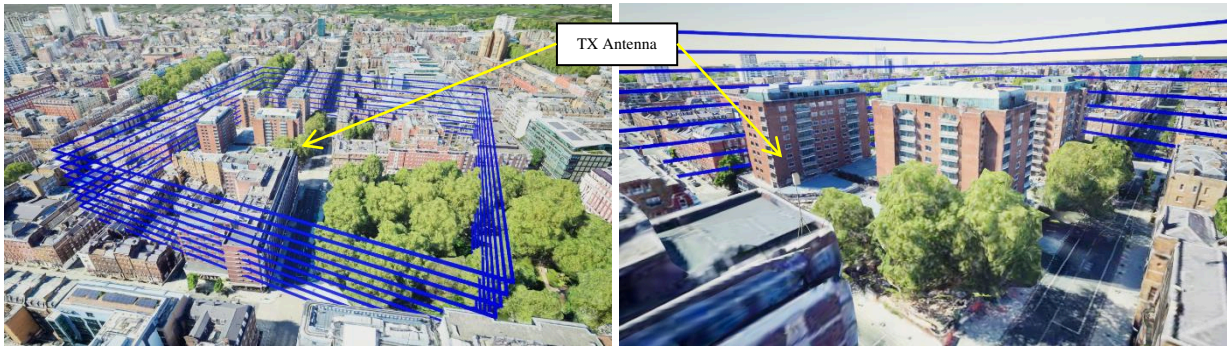
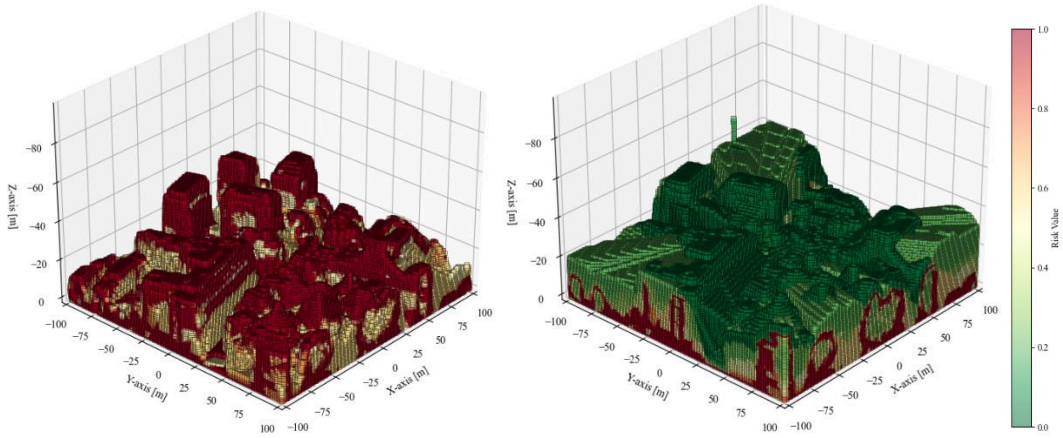


Fig 6. Simulation region within the simulation engine



(a) Voxel risk over 0.5 (b) Voxel risks over 0.0

Fig 7. First party risk map for different risk thresholds in NED coordinates

A. First party risks navigation validation

To properly assess the first party risks model, rather than modelling flights over the buildings level, the trajectories are modelled for highly conflicted scenarios. While altitude considerably increases the impact energy for third party risks, it adds little value to the performance-based risks as these are predominant in low altitude flights, especially through urban canyons. The start (A) and end (B) points are placed at two different locations at different flight levels above street level, and a perimeter following problem is modelled introducing a middle (AB) waypoint as visible in Fig 8. The results illustrate how the airspace becomes less conflicted when increasing the altitude – note how the buildings outline is define but not filled, due to the mesh nature of the 3D map rather than solid geometry, being the inner nodes never reached due to the maximal risk boundary defined around them – and are therefore ignored. The A* path planning algorithm strongly performs the Manhattan distance reduction while avoiding high risk voxels, while the DQL-RL algorithm appears to be more risk sensitive resulting into more immediate risk reduction.

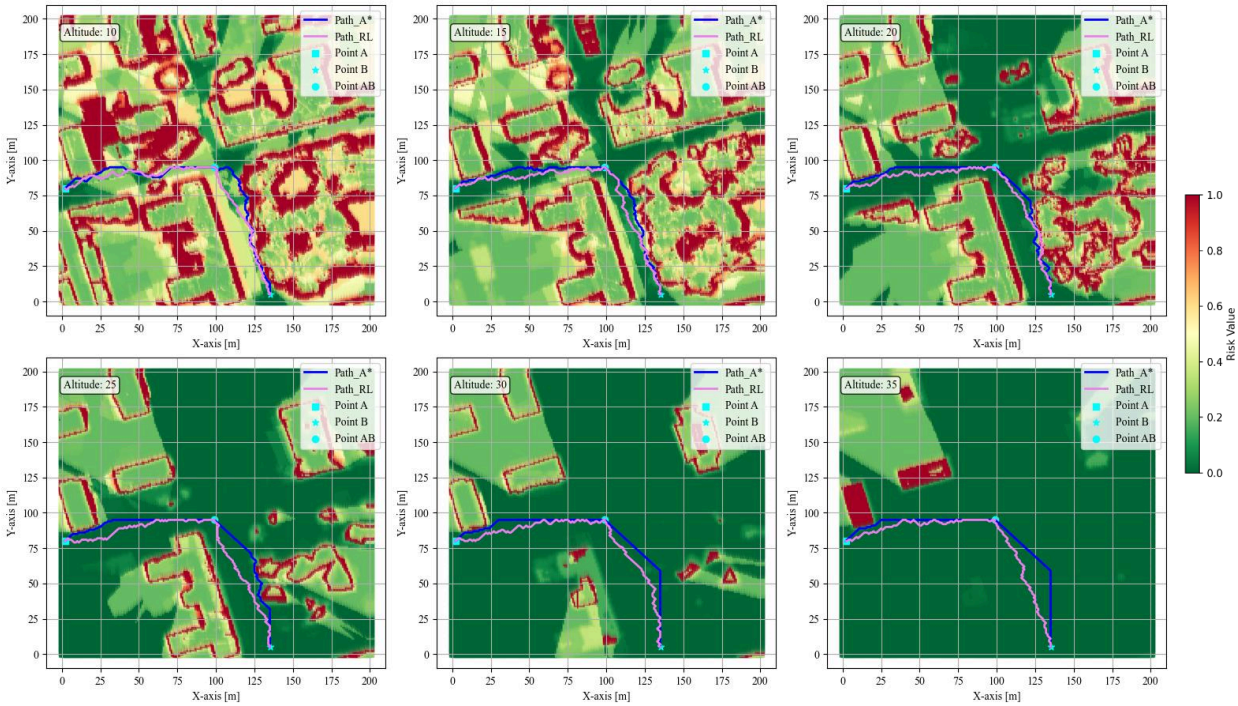
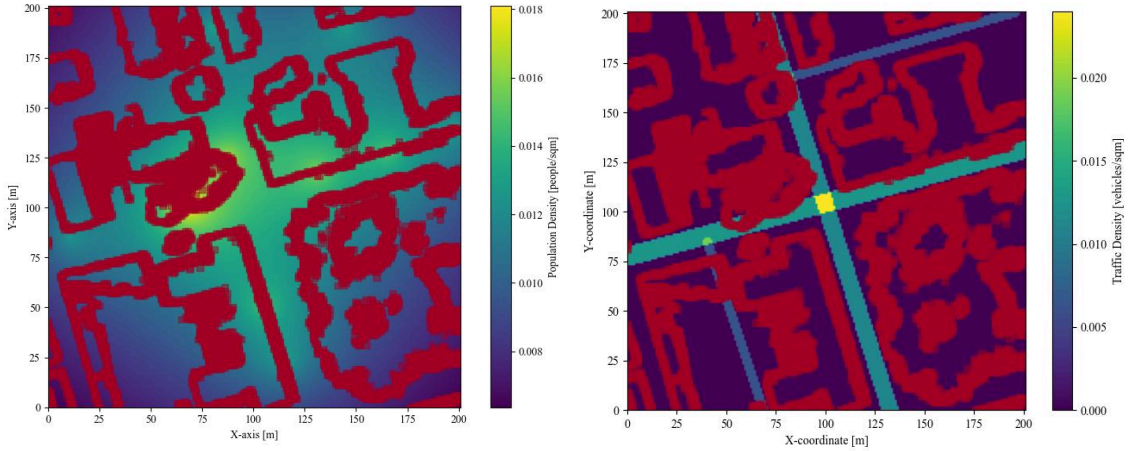


Fig 8. A* path planning and DQL-RL navigation results for different altitudes

First party risks can be used for operations where few urban non-controlled elements are involved, for instance a facility roof inspection in a segregated region from the population and traffic, as an industrial area. Drone performance can be accurately estimated, and no-flight zones are effectively identified and contrasted to the initially planned waypoints and trajectory.

B. Integrated first and third party risks navigation

Third party layers distributions are shown in Fig 9. Expected population density (a) is modelled according to relevant key locations found in the regions, where major hotels can be found, as well as restaurants, bus stops and other high affluence establishments. Note how for the central junction several nodes are concentrated and therefore the distribution is affected. Road traffic (b) concentrates the city traffic onto the roads, where 2 major and 2 secondary streets are found, resulting into higher expected traffic at the junction as well. The third map used for third party risks identifies the vegetation coordinates, corresponding to the park visible in Fig 6, bottom right for the 2D maps. Crash locations outside green areas are classified by crash location altitude, differentiating urban ground areas and vertical elements based on whether the crash coordinates are found under or over 2 meters, respectively.



(a) Population density distribution (b) Traffic density distribution

Fig 9. Population and vehicles density results for the simulation region over projected geometry

The integrated model is tested for a corner-to-corner flight, simulating the fastest path between two points along a planned mission destination, including as many urban elements as possible. The start point is taken in a region highly conflicted in terms of obstacles with several tall buildings, crossing over the main roads junction and ends at the park, from where it would continue to the next waypoint. In Fig 10 four different scenarios are shown, from a more geometrically conflicted – at 25 meters – to an obstacle-clear scene, at 40 meters. For the two lower altitude scenarios the navigation algorithm is unable to get past the obstacles when approaching the target, mainly due to the decreased exploration rate found farther from the start node due to the exploration decay. Risk values show how on some regions first party risks take over and by the junction the opposite happens. The region over the park presents a notable contrast given the low third party cost once the obstacles level is cleared. Table 3. Integrated model DQN-RL paths costs analysisTable 3 includes the metrics for the paths assessment, showing how, as expected, the higher altitude flights present lower costs than lower altitudes, especially once crossed the terminal velocity barrier for which the horizontal velocity component will decrease all the way to zero with higher drop heights. This risk model can also be applied to mission planning for a required set of waypoints, establishing an admissible level of risk and assessing the drone operation feasibility.

Altitude	c_{avg}	c_{min}	c_{max}	c_{total}	Distance [m]
25	0.485	0.301	0.955	-	-
30	0.475	0.278	0.947	-	-
35	0.456	0.232	0.944	238.291	238.29
40	0.382	0.225	0.857	231.463	231.46

Table 3. Integrated model DQN-RL paths costs analysis

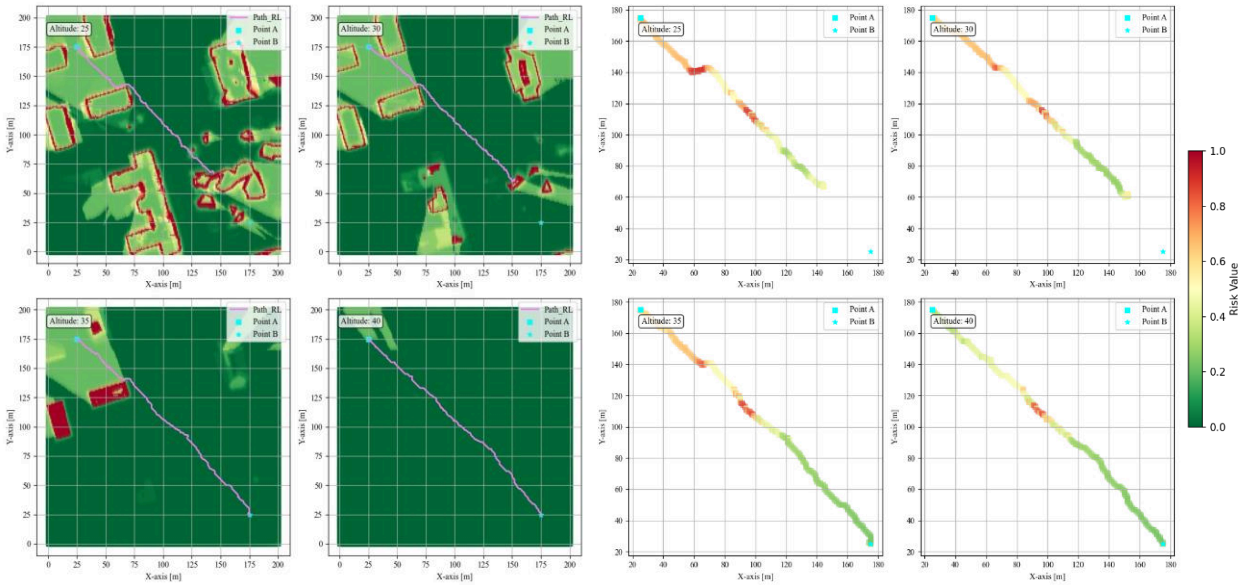


Fig 10. Path trajectories and costs for the integrated risk model at different altitudes

Lastly, different drone paths are modelled to fly across the region and assess the feasibility, setting the main directions parallel to the main roads, as visible Fig 11, for an altitude level of 40 meters. This altitude level implies a reduction of first party risks, while third party risk values are notably influenced by the region the drone flies over. The roads and congested population areas will lead to greater risks that flying over green open areas, for instance. This is also visible for the cost values in Table 4, where for similar path lengths the overall cost almost doubles when travelling along a congested road versus over rooftops and green areas.

Path ID	c_{avg}	c_{min}	c_{max}	c_{total}
1	0.415	0.322	0.661	79.348
2	0.649	0.600	0.856	123.91
3	0.349	0.245	0.595	66.703
4	0.438	0.326	0.708	83.733
5	0.499	0.356	0.756	95.473
6	0.336	0.223	0.712	64.206

Table 4. Flight paths costs assessment

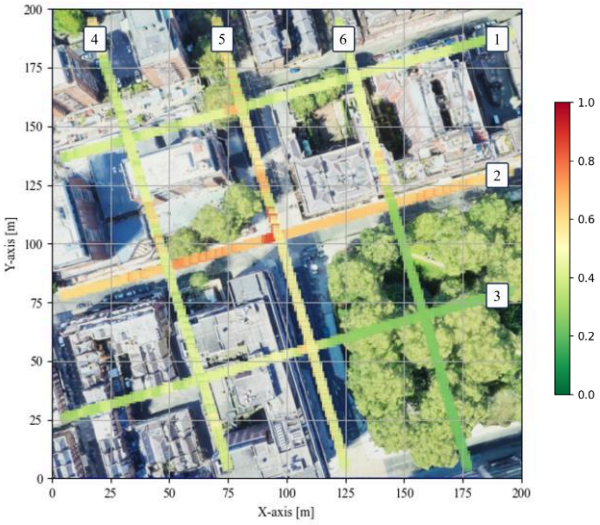


Fig 11. Different flights paths assessment for the integrated risk model at 40m

VI. Conclusions

The modelling of urban environments and the associated operational risks for sUAS is a complex task which requires of accurate geometry modelling as well as a realistic representation of the urban elements. This paper has presented a simulation methodology for realistic urban environments that enables the identification and assessment of sUAS risks in urban environments. Drone motion has been established as a constant level flight in accordance with regular sUAS operations procedures between waypoints, enabling faster flight planning rather than performing continuous ascents and descents from a discrete airspace representation. World geometry meshes are loaded into the simulation engine and serve as the base layer for the risk modelling, incorporating complex urban elements in contrast to statistical and simplified urban models.

A review of the different techniques in risk modelling for sUAS in metropolitan scenes has concluded into a multilayer risk model, accounting for first party drone performance related risks and population, property and societal impact third party risks. Different metrics across the state-of-the-art studies have been integrated to the developed simulation engine, and the resultant risk engine has proven to be an effective tool for urban risks identification, enabling safer sUAS operations planning. Furthermore, the model has been applied to a chosen urban scene, and the obtained results have been validated through A* path planning, DQL-RL and representative waypoints flights mission examples. The modelling of urban population and road traffic provides a clear image of the cost of performing operations over congested areas, while, when possible, the flight over green open spaces will provide safer risk level. First party risk, on the other hand, define no-flight regions for the aircraft, and for a given risk threshold selected by the operator, dangerous regions are visually identifiable and can be easily spotted and avoided. Finally, the model can be tuned to the mission requirements by means of the integrated risk function weights, selecting the most relevant aspects for the mission safety in a given location.

Future work includes the scaling of the model dimensions, which due to the detail precision it might be challenging to process a large urban area within a reasonable time. Additionally, the drone failure descent trajectory can be improved to implement fail modes and add additional motions to the fall. Finally, the path planning and reinforcement learning implemented in this paper aim to provide a use case of how this model can be applied, which could greatly be benefited from increased robustness techniques from smoother trajectories and better awareness of the surrounding risks to avoid unsuccessful trajectories.

Acknowledgments

This work is an industrial research collaboration sponsored by Cranfield University and HEROTECH8.

References

[1] L. E. Alvarez, J. C. Jones, A. Bryan, and A. J. Weinert, ‘Demand and Capacity Modeling for Advanced Air Mobility’, in *AIAA AVIATION 2021 FORUM*, VIRTUAL EVENT: American Institute of Aeronautics and Astronautics, Aug. 2021. doi: 10.2514/6.2021-2381.

[2] T.-H. Tran and D.-D. Nguyen, ‘Management and Regulation of Drone Operation in Urban Environment: A Case Study’, *Soc. Sci.*, vol. 11, no. 10, p. 474, Oct. 2022, doi: 10.3390/socsci11100474.

[3] C. Jiang, H. A. Blom, and A. Sharpanskykh, ‘Third Party Risk Indicators and Their Use in Safety Regulations for UAS Operations’, in *AIAA AVIATION 2020 FORUM*, VIRTUAL EVENT: American Institute of Aeronautics and Astronautics, Jun. 2020. doi: 10.2514/6.2020-2901.

[4] G. Wild, J. Murray, and G. Baxter, ‘Exploring Civil Drone Accidents and Incidents to Help Prevent Potential Air Disasters’, *Aerospace*, vol. 3, no. 3, p. 22, Jul. 2016, doi: 10.3390/aerospace3030022.

[5] J. M. Campbell, ‘Psychological Effects on UAV Operators and Proposed Mitigation Strategies to Combat PTSD’.

[6] D. Vincenzi, D. Ison, and D. Liu, ‘Public Perception of Unmanned Aerial Systems (UAS): A Survey of Public Knowledge Regarding Roles, Capabilities, and Safety While Operating Within the National Airspace System (NAS)’.

[7] V. Sharma, ‘Advances in Drone Communications, State-of-the-Art and Architectures’, *Drones*, vol. 3, no. 1, p. 21, Feb. 2019, doi: 10.3390/drones3010021.

[8] S. Watkins *et al.*, ‘Ten questions concerning the use of drones in urban environments’, *Build. Environ.*, vol. 167, p. 106458, Jan. 2020, doi: 10.1016/j.buildenv.2019.106458.

[9] C. A. Ochoa and E. M. Atkins, ‘Urban Metric Maps for Small Unmanned Aircraft Systems Motion Planning’. arXiv, Feb. 14, 2021. [Online]. Available: http://arxiv.org/abs/2102.07218

[10] B. Pang, X. Hu, W. Dai, and K. H. Low, ‘Third Party Risk Modelling and Assessment’.

[11] X. Ren and C. Cheng, ‘Model of Third-Party Risk Index for Unmanned Aerial Vehicle Delivery in Urban Environment’, *Sustainability*, vol. 12, no. 20, p. 8318, Oct. 2020, doi: 10.3390/su12208318.

[12] X. Hu, B. Pang, F. Dai, and K. H. Low, ‘Risk Assessment Model for UAV Cost-Effective Path Planning in Urban Environments’, *IEEE Access*, vol. 8, pp. 150162–150173, 2020, doi: 10.1109/ACCESS.2020.3016118.

[13] N. Shiode, ‘3D urban models: Recent developments in the digital modelling of urban environments in three-dimensions’.

[14] S. Spiegel and J. Chen, 'Using Simulation Data From Gaming Environemnts For Training A Deep Learning Algorithm on 3D Point Clouds', *ISPRS Ann. Photogramm. Remote Sens. Spat. Inf. Sci.*, vol. VIII-4/W2-2021, pp. 67–74, Oct. 2021, doi: 10.5194/isprs-annals-VIII-4-W2-2021-67-2021.
[15] W. Budiharto, E. Irwansyah, J. S. Suroso, A. Chowanda, H. Ngarianto, and A. A. S. Gunawan, 'Mapping and 3D modelling using quadrotor drone and GIS software', *J. Big Data*, vol. 8, no. 1, p. 48, Dec. 2021, doi: 10.1186/s40537-021-00436-8.
[16] 'HeroTech8 – Fully Automated Drone Services'. [Online]. Available: https://www.herotech8.com/
[17] M. Fu, W. Song, Y. Yi, and M. Wang, 'Path Planning and Decision Making for Autonomous Vehicle in Urban Environment', in *2015 IEEE 18th International Conference on Intelligent Transportation Systems*, Las Palmas: IEEE, Sep. 2015, pp. 686–692. doi: 10.1109/ITSC.2015.117.
[18] J. Wu, H. Wang, N. Li, P. Yao, Y. Huang, and H. Yang, 'Path planning for solar-powered UAV in urban environment', *Neurocomputing*, vol. 275, pp. 2055–2065, Jan. 2018, doi: 10.1016/j.neucom.2017.10.037.
[19] M. Wulfmeier, D. Z. Wang, and I. Posner, 'Watch this: Scalable cost-function learning for path planning in urban environments', in *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Daejeon, South Korea: IEEE, Oct. 2016, pp. 2089–2095. doi: 10.1109/IROS.2016.7759328.
[20] J. Muñoz, B. López, F. Quevedo, C. A. Monje, S. Garrido, and L. E. Moreno, 'Multi UAV Coverage Path Planning in Urban Environments', *Sensors*, vol. 21, no. 21, p. 7365, Nov. 2021, doi: 10.3390/s21217365.
[21] Q. Kuang, J. Wu, J. Pan, and B. Zhou, 'Real-Time UAV Path Planning for Autonomous Urban Scene Reconstruction', in *2020 IEEE International Conference on Robotics and Automation (ICRA)*, Paris, France: IEEE, May 2020, pp. 1156–1162. doi: 10.1109/ICRA40945.2020.9196558.
[22] D. Kamran, C. F. Lopez, M. Lauer, and C. Stiller, 'Risk-Aware High-level Decisions for Automated Driving at Occluded Intersections with Reinforcement Learning', in *2020 IEEE Intelligent Vehicles Symposium (IV)*, Las Vegas, NV, USA: IEEE, Oct. 2020, pp. 1205–1212. doi: 10.1109/IV47402.2020.9304606.
[23] A. T. Azar *et al.*, 'Drone Deep Reinforcement Learning: A Review', *Electronics*, vol. 10, no. 9, p. 999, Apr. 2021, doi: 10.3390/electronics10090999.
[24] 'JARUS – Joint Authorities for Rulemaking on Unmanned Systems'. [Online]. Available: http://jarus-rpas.org/
[25] 'Unreal Engine | The most powerful real-time 3D creation tool', Unreal Engine. [Online]. Available: https://www.unrealengine.com/en-US
[26] S. Shah, D. Dey, C. Lovett, and A. Kapoor, 'AirSim: High-Fidelity Visual and Physical Simulation for Autonomous Vehicles'. arXiv, Jul. 18, 2017. [Online]. Available: http://arxiv.org/abs/1705.05065
[27] 'CelesTrak'. [Online]. Available: https://celestrak.org/
[28] 'Civil Aviation Authority'. [Online]. Available: https://www.caa.co.uk/
[29] 'Federal Aviation Administration'. [Online]. Available: https://www.faa.gov/
[30] 'EASA', EASA. [Online]. Available: https://www.easa.europa.eu/en/landing
[31] G. Balamurugan, J. Valarmathi, and V. P. S. Naidu, 'Survey on UAV navigation in GPS denied environments', in *2016 International Conference on Signal Processing, Communication, Power and Embedded System (SCOPES)*, Paralakhemundi, Odisha, India: IEEE, Oct. 2016, pp. 198–204. doi: 10.1109/SCOPES.2016.7955787.
[32] N. Gyagenda, J. V. Hatilima, H. Roth, and V. Zhmud, 'A review of GNSS-independent UAV navigation techniques', *Robot. Auton. Syst.*, vol. 152, p. 104069, Jun. 2022, doi: 10.1016/j.robot.2022.104069.
[33] H. Azami, M.-R. Mosavi, and S. Sanei, 'Classification of GPS Satellites Using Improved Back Propagation Training Algorithms', *Wirel. Pers. Commun.*, vol. 71, no. 2, pp. 789–803, Jul. 2013, doi: 10.1007/s11277-012-0844-7.
[34] 'Unmanned Aircraft System Operations in UK Airspace – Operating Safety Cases - CAP722', 2022.
[35] A. Ferlini, W. Wang, and G. Pau, 'Corner-3D: a RF Simulator for UAV Mobility in Smart Cities', in *Proceedings of the ACM SIGCOMM 2019 Workshop on Mobile AirGround Edge Computing, Systems, Networks, and Applications*, Beijing China: ACM, Aug. 2019, pp. 22–28. doi: 10.1145/3341568.3342108.
[36] F. Pollet, S. Delbecq, M. Budinger, and J.-M. Moschetta, 'Design Optimization of Multirotor Drones in Forward Flight'.
[37] 'Demography, London's Population & Geography', Trust for London. [Online]. Available: https://trustforlondon.org.uk/data/geography-population/
[38] Z. Svatý, L. Nouzovský, T. Mičunek, and M. Frydrýn, 'Evaluation of the drone-human collision consequences', *Heliyon*, vol. 8, no. 11, p. e11677, Nov. 2022, doi: 10.1016/j.heliyon.2022.e11677.
[39] K. Cussen, S. Garruccio, and J. Kennedy, 'UAV Noise Emission—A Combined Experimental and Numerical Assessment', *Acoustics*, vol. 4, no. 2, pp. 297–312, Mar. 2022, doi: 10.3390/acoustics4020018.
[40] S. Park and K. Lee, 'Developing Criteria for Invasion of Privacy by Personal Drone', in *2017 International Conference on Platform Technology and Service (PlatCon)*, Busan, South Korea: IEEE, Feb. 2017, pp. 1–7. doi: 10.1109/PlatCon.2017.7883702.

Risk assessment for sUAS in urban environments: a comprehensive analysis, modelling and validation for safe operations

Celdran Martinez, Victor

2024-01-04

Celdran Martinez V, Shin HS, Tsourdos A. (2024) Risk assessment for sUAS in urban environments: a comprehensive analysis, modelling and validation for safe operations. In: AIAA SCITECH 2024 Forum, 8-12 January 2024, Orlando, USA. Paper number AIAA 2024-0232
https://doi.org/10.2514/6.2024-0232