Fourteenth USA/Europe Air Traffic Management Research and Development Seminar (ATM2021)

Ground risk vs. Efficiency in Urban Drone Operations

Leonid Sedov Valentin Polishchuk
Communications and Transport Systems, ITN,
Linköping University, Norrköping, Sweden

Vishwanath Bulusu
Crown Consulting Inc,
Moffett Field, California, USA

***Abstract*—This paper explores tradeoffs between ground impact and efficiency of drone flights in urban scenarios. We give an algorithm which produces a set of routes with different lengths and varying number of people affected by the drone. We also present an interactive online visualization tool allowing the user to modify flightpaths in order to explore routing options. Our path finder and the GUI are implemented for a metropolitan area of Norrköping municipality in Sweden. The methods studied in this paper may give UTM service provider the tools to negotiate flightplans which will be acceptable by both the regulator and the drone operator.**

***Keywords*—Urban Airspace; Ground Risk; Flight Efficiency**

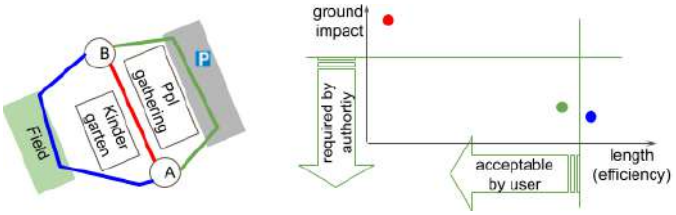

Figure 1. A conceptual illustration of possible flight paths output by our algorithm: red path is short but may affect many people; blue path has very low risk but is too long, and green has ground impact low enough for approval of the operation and is sufficiently enough to be viable for the operator

I. Introduction

The importance of using low-ground-impact paths for UAV (Unmanned Aerial Vehicle) operations in urban scenarios has been widely acknowledged [1]–[5]. While in conventional aviation, risk management focuses primarily on prevention of hull loss and associated risks for the vehicle occupants (1st party risks), for UAV missions it is vital to minimize the number of people *on the ground* who can be potentially affected by a dysfunctional drone overflying populated areas (3rd party risks): the loss of a pizza delivery drone by itself is not a catastrophe. The challenge is that in metropolitan regions, maximizing route efficiency (the goal of the user) and minimizing ground risk (the concern of the authorities) often are *conflicting* objectives: the direct (most efficient) route may straddle through spots of high population density, while a path through unpopulated areas may be unduly long (for a real-world snapshot of a direct and an alternative route see e.g., MEDUSA [6, Section 7.1.5] methodology example).

This paper presents an algorithm (Section II) to compute a collection of UAV flightpaths, between a given origin–destination pair (Fig. 1), which trade off ground impact (expressed as the affected population) and efficiency (path length). Each of the produced paths has minimum length for a given number of affected people and affects fewest people for its length (i.e., we output so called *Pareto optimal* solutions). By working out formulas for the expected fatality rate (EFR) of the operations (Section I-A), we show that our paths are Pareto-optimal with respect to EFR and length as well (Lemma 1). On the experimental side (Section III), we present a web-based GUI for the user to click in a path and modify it, observing the changes in the ground impact and path length. We demonstrate the output of our algorithm and the GUI on population data for Norrköping municipality in Sweden. Finally future work is discussed in the conclusion (Section IV).

A. Preliminaries: Linear density as Safety level

One motivation for our work comes from the vision that the expected number of fatal injuries to 3rd parties on the ground (per flight hour or per takeoff) has been identified as one of the most important metrics for ground risk. The number of fatalities is directly proportional to the number of people who may be affected in case the UAV falls down from the skies. The coefficient of the proportionality consists from several factors – vehicle reliability, probability of fatality after impact with a person (which, in turn, depends on the vehicle mass, presence of a parachute, etc.) and others. Determining the factors and quantifying their values for different drones is a rich research area in itself [4]–[17] and is outside the scope of this paper. We only use the fact that irrespective of what the risk-contributing factors are, if their values are fixed, then minimizing the number of fatalities is equivalent to minimizing the affected population. This observation allows us to decouple flight planning from vehicle characteristics and focus just on routing the paths for a design aircraft. The paths, output by our algorithm, may then be coupled with vehicle-specific factors

in order to decide which paths are admissible to fly by which drone models. These routing options may be fed into SORA, MEDUSA and further up the risk assessment workflow, to determine the safety parameters of the operations. Overall, it is envisioned that the target level of safety for UAV operations will have to match that of the conventional aviation [8]–[10], [18]–[20]. Our methods may provide a stepping stone towards reaching and confirming such Equivalent (or Equal) Level of Of Safety (ELOS) for drone traffic.

Observations from the previous paragraph may be put into formal quantitative terms by writing out formulas for the Expected Fatality Rate (EFR). For concreteness, consider the Arctic Science RPAS Operator's Handbook [21, page 19] which bases its formulas on those in the seminal work of Weibel and Hansman [22, Equations (1) and (2)] (an "executive summary" of the formulas can be found in Figure 5 of SORA1.0). The ground risk formulas from [21], [22] essentially formalize the following fact: if every time that a failure happens, an expected number N_{exp} of people are exposed to the hazard (falling debris, etc.) and an exposed person dies with probability P, then

$$EFR = \frac{N_{exp} P}{MTBF} \tag{1}$$

where $MTBF$ is the mean time between failures.

As mentioned above, we assume that P and $MTBF$ are fixed (they depend on the vehicle, mitigation measures, shelters, etc.): our focus is on path planning, and the only term in Equation (1) depending on the path (on population under the path) is N_{exp}. Earlier work estimated N_{exp} via $N_{exp} = A_{exp} \cdot \rho$, where A_{exp} is the "lethal debris area" [22] and ρ is the population density. The area A_{exp} is proportional to d^2 where d is the distance from the drone reached by the debris (e.g., according to the 1:1 rule from EASA's guidance [23, Section 2.3.1(c3)], d is at least the planned flight altitude; [4] showed that in some scenarios the impact area can be even larger). We are not concerned with the exact shape of the impact region – it can be e.g., the radius-d disk ($A_{exp} = \pi d^2$), or $2d \times 2d$ square ($A_{exp} = 4d^2$) centered on the drone. Since EFR is estimated to the order of magnitude, we set $\pi \approx 4$ (notwithstanding Geometry teachers complaints) and use $A_{exp} = 4d^2$ (our results do not depend on the constant in front of d^2).

The *crucial difference* of this paper from prior work os that earlier the population density ρ was assumed *constant* (to quote [22], "The probabilistic expectation assumes that the population is evenly distributed over the area of interest."). The assumption of uniform density implied, in particular, that N_{exp} did not depend on the drone path – hence the focus of prior work on P and $MTBF$ (mitigation and vehicle reliability), and not on route planning.

In this paper we lift the uniform-density assumption and compute the total number of people N_{tot} potentially exposed to the hazard – i.e., the population under the width-$2d$ strip centered on the path (see, e.g., Fig. 8). If the path has length L, then the strip has area $2dL$. Assuming that the failure is equally likely to occur at any point along the path, the expected number of people in the exposed region (area $A_{exp} = 4d^2$ chosen uniformly at random along the path) is $N_{exp} = N_{tot} \frac{4d^2}{2dL} = 2N_{tot} d/L$ (Fig. 2). Substituting this into Equation (1), we obtain that for a given d

$$EFR = \text{const} \cdot \frac{N_{tot}}{L} \tag{2}$$

where const is some value which we do not influence. This motivates finding various-lengths paths with various N_{tot} (and N_{tot}/L) – the subject of this paper.

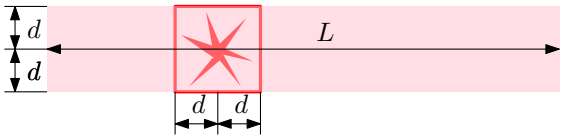

Figure 2. If $2d \times 2d$ square sliding along length-L width-$2d$ strip stops at a random point, the expected number of people in the square is $N_{tot} \frac{4d^2}{2dL}$

B. Related work

According to DeepBlue [24], one important innovation brought by EASA's EU drone regulation [20] is that different operations will have to meet different requirements, even if executed by the same drone. Similarly, the same route may be safe to fly by one drone but too risky for another, e.g., heavier, vehicle. Since the probability of fatal injury depends on the drone's kinetic energy, our algorithms may be used to define the speed limits for drones of different mass in different areas, depending on the population density. Our paper thus contributes to definition and use of PBN for UTM [17], [25], by following Mantra2 "Risk-based approach where geographical needs and use cases determine the airspace performance requirements" from NASA's original UTM ConOps [26] and one of the key principles in SESAR's U-space Blueprint [27] encouraging "To follow a risk-based and performance-driven approach when setting up appropriate requirements..."

Many works observe that risk-based geographical planning crucially relies on knowing the population density [6], [20], [28], [29]. For more realistic results, a dynamic map would be more appropriate, but in the absence of reliable real-time population mobility data, using a static map (as we do in this paper) could be considered as a starting point [30], [31] (the density does not change much during the 10-30min of the drone flight time). This is inline with the phased research approach outlined by MITRE [8], going from mapping mission profiles to risk variables (Phase 1) to incorporating risk-minimization into flight planning (Phase 2) to identifying the risks in real time (Phase 3). In this sense, our GUI addresses the route-specific part of the first phase and our algorithms – of the second phase; Phase 3 is left for future work.

The current special operations risk assessment from (SORA) [10] is semi-qualitative, as it associates the risks with integers 1–7 based on pre-defined operational scenarios. Supporting digitalization of risk assessments was named one of the "immediate hard challenges facing UTM" by GUTMA [32].

In particular, the need to digitize SORA's ground risk classes (GRC) assignment was stressed during Riga airspace assessment [31]. Our GUI is similar to the conceptual example for SORA Tool from [31, Section 5.4.3], but displays continuous values of the KPIs instead of the discrete GRCs. More generally, our results help with the automation of risk assessment, by quantifying the risks of overflying the population instead of dealing with more qualitative notions of densely/sparsely populated areas.

Multicriteria optimization is omnipresent in ATM research, owing to the multitude of aviation stakeholders, often having conflicting objectives. The explored pairs of conflicting objectives range from flight time vs. fuel consumption [33] to user-preferred routes vs. flow controllability [34] to CO2 emissions vs. noise exposure [35]. Since this paper presents an abstract treatment of path length vs. ground impact tradeoffs, our methods can be used also to study the noise impact (e.g., the number of people exposed to certain dB levels) instead of the ground risk. Earlier, noise impact of emerging operations in urban scenarios was studied in [36]–[40]. Heuristics for minimizing the weighted sum of path length and ground risk were explored in [41]; the edges of the output paths were following 8 orientations which made the paths look somewhat unnatural, even after local improvements (this could be attributed to the fact that heuristics do not give any optimality guarantees).

II. Modeling and solution

In the input to our problem we have population data, specifying the number of people in every gridpoint over a metropolitan area (refer to Fig. 8 below). Two gridpoints s, t are designated as the start and the end points of the sought path. We are also given the width d of the impact area around the drone: the drone path is thickened by d, and anybody in the area covered by thickened path can be struck by the drone.

We form a complete graph $G = (V, E)$ on the gridpoints (i.e., the vertices V are the gridpoints and the edges E connect every pair of vertices). The *length* l_{ij} of the edge $ij \in E$ between vertices $i, j \in V$ is the Euclidean distance between i and j. In addition, we define the *weight* w_{ij} of the edge to be the total number of people who fall inside the thickened edge, i.e., the width-$2d$ rectangle having ij as the midline. Refer to Fig. 8, left and middle: in each of them, the one-edge path is an example of such a thickened edge. Note that the weight depends on d, but in a given instance of the problem (our focus in this section), d is fixed and so is w_{ij} (for our GUI, in which d is a user-adjustable parameter, we precompute the weights for a range of d values so that the GUI instantaneously reacts as the user moves the slider for d, but in this section d is fixed).

For an s-t path π in G, the length of the path (the total length of its edges) is denoted by $l(\pi)$, and the weight of π (the total weight of edges) by $w(\pi)$. The path π is called *Pareto optimal* if it has smallest weight for a given length and it is shortest for its weight, i.e., if one objective cannot be improved without deterioration of the other (formally, π is Pareto optimal if any other s-t path π' with $w(\pi') < w(\pi)$ has $l(\pi') > l(\pi)$, and any other s-t path π' with $l(\pi') < l(\pi)$ has $w(\pi') > w(\pi)$). The set of all Pareto optimal paths is called the *Pareto frontier* (Fig. 3).

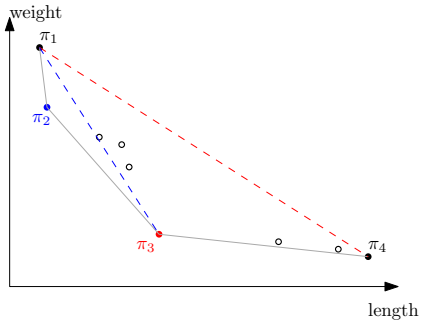

Figure 3. Circles give example lengths and weights of Pareto-optimal paths. $\pi_1, \ldots, \pi_4$ are the paths on the lower hull of the Pareto frontier (the hull is shown with the thin line). π_3 is the extreme point of the frontier in the direction perpendicular to the segment between the extreme Pareto optimal solutions π_1 and π_4. To find the other points on the lower hull, we recurse on both sides of π_3: on the right no new solutions are found because both π_3 and π_4 are extreme in the direction perpendicular to the segment $\pi_3\pi_4$; on the left, a new Pareto optimal path π_2 is discovered as the extreme point in the direction perpendicular to $\pi_1\pi_3$.

Recall from Equation (2) that the EFR of a path is proportional not just to the potentially exposed population (N_{tot} in Equation (2)), but to the population per length of the path (N_{tot}/L). Nevertheless, the next lemma shows that it suffices to focus only on Pareto-optimal paths in terms of L and N_{tot}:

Lemma 1. *Pareto-optimal paths in terms of the length and the number of potentially exposed people (i.e., the Pareto frontier in the (L, N_{tot})-plane) are also Pareto-optimal in terms of the length and the EFR (i.e., the same paths appear on the Pareto frontier in the $(L, N_{tot}/L)$-plane).*

Proof. Let π be a path on the Pareto frontier in the (L, N_{tot})-plane. If there existed a competitor path π' for which both L and N_{tot}/L were smaller than those for π, then also the product $LN_{tot}/L = N_{tot}$ would have been smaller for π'. That is, both L and N_{tot} would have been smaller for π', contradicting Pareto-optimality of π. □

A. Computing Pareto optimal solutions: the theory

The Pareto frontier can be found as follows. Let $W = w_{st}$ be the weight of the shortest s-t path in G (which is the edge st). Since in our problem the weight of every edge is an integer (the number of exposed people), every edge $ij \in E$ can be replaced by a path of w_{ij} unit-weight edges. In the obtained graph $G' = (V \cup V', E')$, the vertex set consists from the vertices V of G plus the vertices added along the edges of G. The length of every edge in G', obtained from an edge $ij \in E$, is equal to l_{ij}/w_{ij}. The weight of every edge in E' is 1.

The reason to create G' is that the length of an s-t path in G is the same as in G', while the weight of the path in G is equal to the number of edges of the path in G'; in particular, the largest number of edges in an s-t path in G' is W. Thus,

if for every integer $k \leq W$ we know the length of the shortest s-t path with $\leq k$ edges in G', we know the length of the shortest path with weight $\leq k$ in G – which is exactly the Pareto optimal paths (our Holy Grail).

Finally, shortest $\leq k$-edge s-t paths in G' can be found for all k by a modification of Bellman–Ford dynamic programming algorithm. Specifically, let $dist(i,k)$ denote the length of the shortest $\leq k$-edge path from s to vertex $v \in V'$; we call it the *label* of the vertex. Initialize the labels $dist(v,k) = \infty$ for all v, k, except for $dist(s,0) = 0$, The algorithm proceeds by computing the labels for increasing k. For every vertex v, the shortest $\leq k$-edge path from s is either the shortest $\leq (k-1)$-edge path to v (having an extra edge does not help), or is the shortest $\leq (k-1)$-edge path from s to a neighbor u of v plus the edge uv:

$$dist(v,k) = \min[dist(v,k-1), \min_{u:uv \in E'} dist(u,k-1) + l'_{uv}]$$

where l'_{uv} is the length of edge $uv \in E'$ (recall that edges of G' are inherited form G and that the length of each edge inherited from an edge $ij \in E$ is l_{ij}/w_{ij}).

Unfortunately, for our problem size, running the above algorithm is computationally infeasible. First, our population grid is 250x250, so the number of vertices of G is 62500, and the number of edges in the complete graph G is quadratic in that. This makes computing even a single shortest path in the graph problematic. In addition, the weight of the shortest path W (the number of affected people) can be as large as $\sim$11000, so maintaining the labels for all possible $k \leq W$ would take too much space. In what follows, we present practical ways to cope with the size of our problem.

B. Practical solutions

Let δ denote the granularity of our population map, or, equivalently, the distance between the closest vertices in V. To decrease the size of the graph searched for the paths, we do not connect every vertex $v \in V$ to all other vertices, Instead, we choose a number $r \geq 2\sqrt{2}\delta$ and connect v only to the vertices within distance r from v; let $G_r = (V, E_r)$ denote the graph so obtained. The next lemmas bound the approximation error of computing paths in G_r instead of G.

Lemma 2. *For every edge $ij \in E$ there exists a path π_{ij} in G_r, whose length is at most $(1 + 2\sqrt{2}\delta/r)l_{ij}$.*

Proof. Let $r' = r - \delta\sqrt{2}$, and consider the set S of $\lfloor l_{ij}/r' \rfloor$ points regularly placed on ij at spacing r' (Fig. 4, left). Since V is a regular square grid with granularity δ, for any point $s \in S$ there exists a point $v_s \in V$ within distance $\delta/\sqrt{2}$ from s. For each $s \in S$, cut ij at s and add the connections s-v_s-s from ij and back. The obtained path has length at most $l_{ij} + \delta\sqrt{2}|S| = l_{ij} + \delta\sqrt{2}\lfloor l_{ij}/r' \rfloor \leq (1 + 2\sqrt{2}\delta/r)l_{ij}$. Moreover, since the distance between consecutive points $s, s' \in S$ is at most r' (the distance is exactly r' for all consecutive points except, possibly, for $s' = j$, when the distance may be strictly smaller than r'), the distance $|v_s v_{s'}| \leq |v_s s| + |ss'| + |s' v_{s'}| \leq r$. Thus, v_s and $v_{s'}$ are connected in G_r ($v_s v_{s'} \in E_r$), implying the statement of the lemma. □

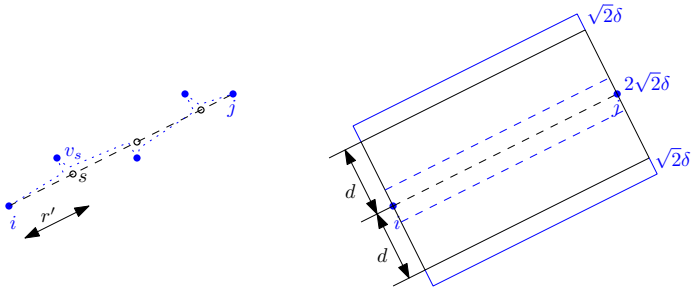

Figure 4. Left: ij (dashed) is approximated by path in G_r (dotted blue, after shortcutting). Filled blue circles are the gridpoints (points of V) closest to the points of S (hollow circles). Right: Since the path lies within $\sqrt{2}\delta$ of ij (blue dashed), the impact area of ij (black) dilates by at most $\sqrt{2}\delta$ (blue).

We now turn to bounding how well G' approximates the weights of G. We assume that the population density is *Lipschitz-continuous*, i.e., that it does not grow "too sharply". Recall that a function $f : \mathbb{R}^2 \mapsto \mathbb{R}$ is *α-Lipschitz* if $\forall p, q \in \mathbb{R}^2, |f(p) - f(q)| \leq \alpha|pq|$; the minimum α for which the inequality holds is called the *Lipschitz constant* of f. The notion of Lipschitz continuity generalizes the usual continuity to non-differentiable functions (for a differentiable f, the Lipschitz constant is just the bound on $|\nabla f|$). Assuming Lipschitz continuity of the population density f is reasonable because there is an upper bound on how crowded people on the ground can be, and abrupt jumps of f are due to high-rise buildings where people are sheltered from UAVs immediate impact (if this is not the case, a high-density spot may be declared an obstacle, by removing the graph edges in the radius-d around the spot, so the spot is not affected by computed flightpaths).

Lemma 3. *For every edge $ij \in E$ there exists a path π_{ij} in G_r, which affects at most $w_{ij}(1 + \sqrt{2}\delta/d)$ people and whose length is at most $l_{ij}(1 + 2\sqrt{2}'a\delta/r)$.*

Proof. Let R be the width-$2d$ rectangle having edge $ij \in E$ as the midline (Fig. 4, right); the weight w_{ij} of the edge is the number of people inside R. It follows from the proof of Lemma 2 that the path π_{ij} in G_r, approximating ij, lies within distance $\delta/\sqrt{2}$ from ij. Thus, the area affected by the path lies inside the width-$(2d + 2\sqrt{2}\delta)$ rectangle R' centered on ij. By Lipschitz-continuity of f, the number of people affected by π_{ij} is at most $(w_{ij}/2d)(2d + 2\sqrt{2}\alpha\delta)$. □

Lower hull of Pareto frontier

As noted above, since the weight W of the shortest s-t path in G (or equivalently, the number of edges in the shortest s-t path in G') is large, we cannot afford to store the distance labels for all possible path weights and cannot compute all Pareto optimal solutions. Instead, we compute the points on the *lower hull* of the Pareto frontier (refer to Fig. 3). Any path on the hull minimizes a linear combination $l(\pi) + \beta w(\pi)$ of path length and weight for some β. This is because the path is the first point of the Pareto frontier hit by the line $l + \beta w = C$ as it moves up when C increases. Hence, any path on the hull

may be found by finding the (usual, single-criteria) minimum-cost s-t path in the graph whose edge costs are $l + \beta w$ (that is, the cost of each edge ij is $l_{ij} + \beta w_{ij}$).

We thus search for the paths on the hull recursively: Find the minimum-length and minimum-weight paths π_1 and π_{end} resp., and determine β from the slope of the segment connecting the paths. Set the costs to $l + \beta w$ and compute the minimum-cost s-t path (equivalently, find the first point hit by the line $l + \beta w = C$). If no point is found (i.e., if any of π_1, π_{end} is a minimum-cost path), terminate – the paths π_1, π_{end} are the only paths on the hull. Otherwise, recurse on the both sides of the newly found path π: check whether there is another extreme point on each side of π (if found, recurse further, until no new extreme point is found).

III. Experiments and GUI

We implemented our algorithm and report on the results below.

A. Synthetic data

First, we experimented on a small 50×50 instance (Fig. 5), using a simple artificial population density map: the whole square had low density, except the 15×30 rectangle with very high density (purple in Fig. 5, left and middle). We ran both the exact algorithm to find the full Pareto frontier of paths and the practical algorithm to find the paths which comprise the lower hull of the Pareto frontier (we used $d = 5$ pixels). Figure 5, right shows the Pareto frontier (red and blue asterisks) and its lower hull (red asterisks). It can be seen that while theoretically the full Pareto frontier contains more paths to choose from, from the practical point of view the lower hull can also provide enough different choices of flightpaths, effectively removing slightly different variations of the same path.

B. Real data

Next, we tested our algorithm on an instance of Norrköping municipality in Sweden, using data from [42]. The population density was specified on a 250×250 grid with granularity $\delta \approx 13m$. We ran our algorithms with $d = 120$m (a slow light drone) and with $d = 300$m (a drone with more kinetic energy and thus larger lethal area). As discussed at the end of Section II-A, running the exact algorithm to find the whole Pareto frontier is impractical for our instance size, so we only computed the lower hull.

Figure 6 shows the paths obtained for the two values of d (the population map, s, t, etc. are the same for both instances). Note that while for $d = 300$m (Fig. 6, right) the minimum-impact path goes around the whole densely populated city area, the less-impactful drone ($d = 120$m, Fig. 6, left) is able to shortcut between the two high-density spots without exposing too many people to danger; in fact, flying around the whole city with a less-impactfull UAV exposes more people than shortcutting.

It can also be seen that the paths on Figure 6, top right, form 3 clusters, which, informally speaking, represent three "topologically different" route configurations (if we look at the areas with high population density as "holes"). Figure 7 shows the 3 paths that are marked with red asterisks on Figure 6, top right. These 3 routes essentially represent the three viable choices – the other paths on the lower hull of the Pareto frontier are just slightly perturbed versions of these 3 flightpaths. Depending on the flying conditions, planned mission specifics, laws and regulations, it might be better to either choose the route with the minimum length and just quickly fly above people, or the route that goes slightly longer, but squeezes in between two high-density zones, or the longest path which goes around the whole populated part of the city and thus affects much fewer people than the others.

We note that in general it may happen that not all of the paths generated by the algorithm are practically suitable for drone flights (due to, e.g., sharp turns or excessively many turns) and therefore may require additional post-processing (such as, e.g., smoothing). While such a smooth version of a flightpath technically should be worse than the original (since it is not present on the Pareto frontier), it is possible that from the practical standpoint it can be beneficial to simplify a flightpath at the expense of a minor technical increase in the risk estimate.

All experiments were run on a desktop PC with 40GB RAM and 6 CPU cores (12 threads). Finding all paths on the lower hull of the Pareto frontier for one real-world instance took $\sim$30-40 minutes. We believe that the runtime may be decreased by employing multi-threading (our implementation was using only one CPU core most of the time). Most of the computational time was spent on building the graph G_r and updating the edge costs $l + \beta w$ for each value of β. Computing these values on-demand may improve the runtime as well.

C. Interactive visualization

Our experiments above were run with fixed input (s, t, d, etc.). To explore the solution space further, we implemented a GUI [1] that allows the user to click in their own flightpath (including s and t). The user can then modify the path as they wish, adding or moving the path vertices. In addition, we provide the slider for d, changing the affected area thickness. Our implementation recomputes the number of affected people and the path length in real time as the user makes their changes. For visualization purposes, our GUI supports switching between map view and population density view.

Figure 8 shows snapshots of the GUI with several paths.

IV. Conclusion

We gave an algorithm to compute a set of drone routes that differ in length and the number of exposed people; we also presented a GUI for the user to play with their own routes. Our tools may help the sUAS (small unmanned aircraft systems) stakeholder community with creation of a streamlined, repeatable approach for mapping mission profiles to key risk variables (use case-based risk assessment stemming from operational characteristics).

[1] We invite the reader to check it at https://undefiened.github.io/ground_risk/

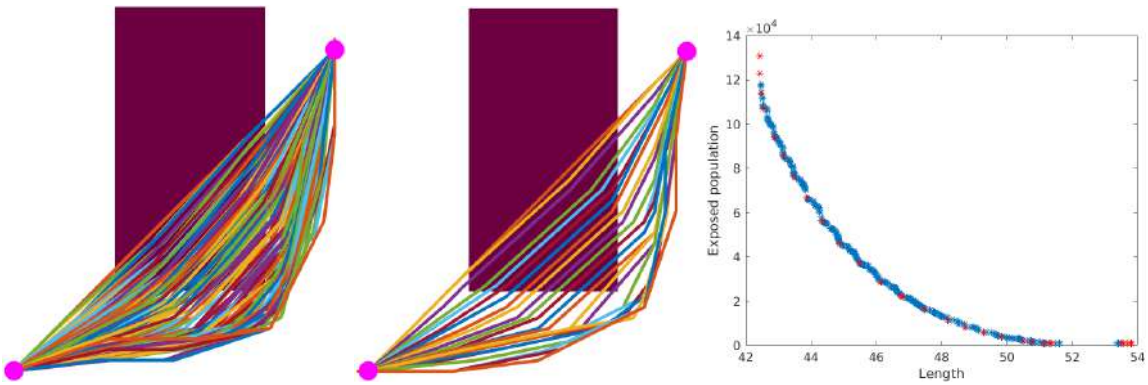

Figure 5. Left: all routes on the Pareto frontier (red and blue asterisks). Middle: routes on the lower hull of the Pareto frontier (red asterisks). Right: the Pareto frontier; the red asterisks mark the routes that form the lower hull, i.e., routes obtained from the practical algorithm.

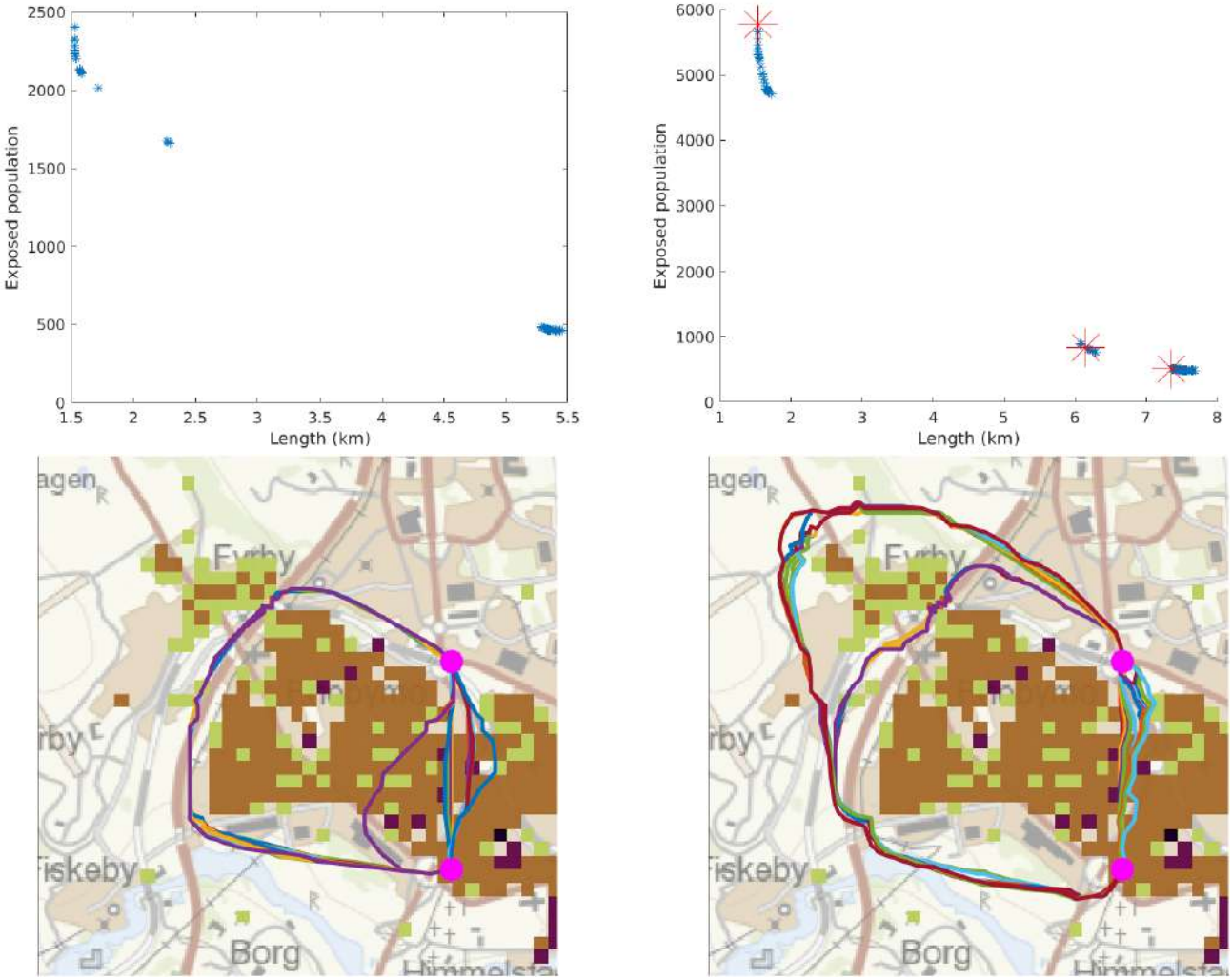

Figure 6. Routes on the lower hull of the Pareto frontier. Left column: $d = 120m$. Right column: $d = 300m$. Top row: the lower hull of the Pareto frontier. Bottom row: the routes. Red asterisks on the top right subfigure mark the routes shown on Figure 7.

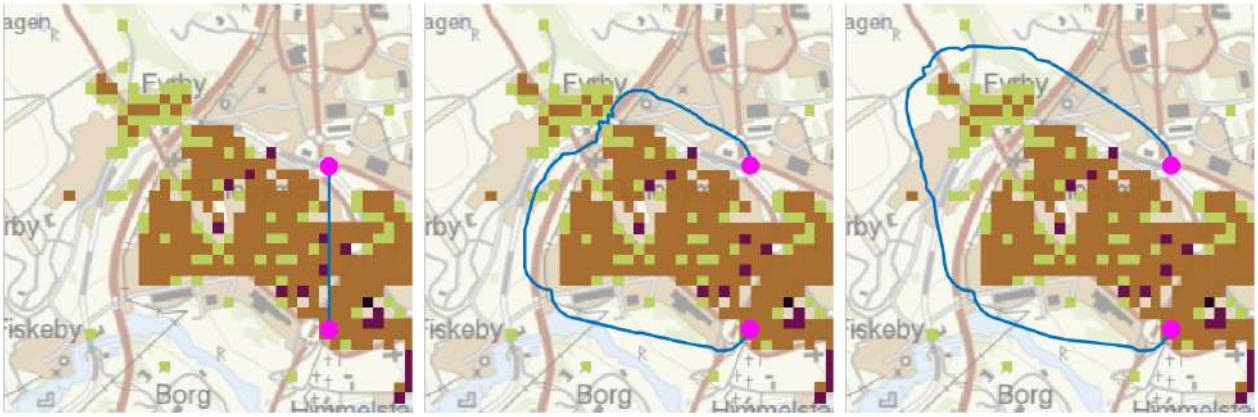

Figure 7. From left to right: paths of increasing length but decreasing weight.

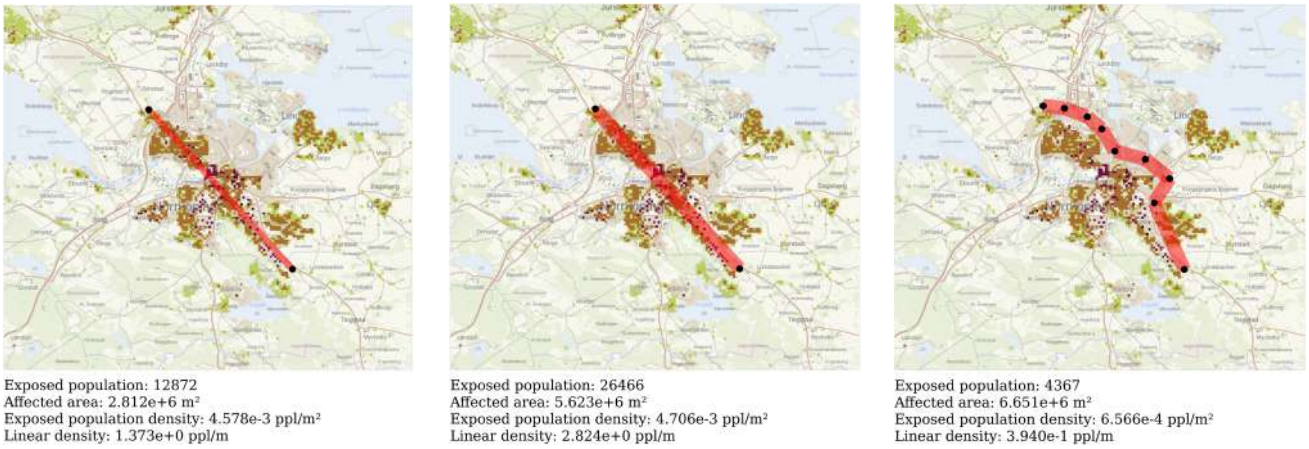

Figure 8. Snapshots of the GUI: the heatmap is the population density, the red is the impacted area. Left: A drone impacting small area around the path (small d). Middle: Larger d. Right: A modified route is longer but exposes fewer people.

Two most outstanding research directions left open by our work are differentiating between people indoors/outdoors and considering *dynamic* population. For the former, the population density map should be enhanced with information about people's vulnerabilities in different parts of the city. Such maps may be obtained from the buildings data which in addition to the standard, concrete constructions will provide information about lighter, possibly temporary, structures like, e.g., canopies or tents for outdoor seating. Note that CORUS ConOps [43, Sec. 3.1.2] lists population map as an "Optional" service in X and Y volumes and "When-Available" in Z volume of U-space; providing the service may be most valuable if it gives the buildings information along the lines outlined above. (This would be inline with aviation's best practices, e.g., w.r.t. weather forecasts – generic products may be used for starters, gradually replaced by services specifically tailored to the aviation industry needs.) In any case, the refined maps may be easily digested by our algorithms, simply via reducing the weight of the pixels where people are sheltered from the failing drones.

For the dynamic population, an extended mobility module attached to our methodology, may account for people in cars, buses and other ground transportation vehicles: when a road stretch falls into the zone potentially affected by the drone crash, a multitude of new parameters may be directly taken into account by adjusting the weights of the pixels through which the road goes. We leave the details to be worked out in the extended module: the speed of the car increases the weight while other factors like traffic intensity, cars windshield resistance et al. may decrease expected number of affected people and hence the weight of the pixels.

Another direction for future research is probabilistic modeling of impact area [12], [16] replacing our fixed-width strip around the flightpath; such modelling may also take into account that faster/higher flying drones affect a wider area. Also, cumulative effects of large-scale drones operations must be taken into account: it would beat the purpose of risk-based routing if a low-density suburban area gets all the traffic (avoiding a higher-density city center), thus endangering the suburban population in the long run; perhaps our work may be extended to risk-averse *flow* routing. Last but not least, for a complete assessment our ground risk estimates should be complemented with air risk calculations. Combined with the SRM (Safety Reference Material), the risk assessment may

feed into the holistic MEDUSA methodology [6].

Acknowledgements. We thank Vijay Augustine (ENAC), Attila Takacs and Richard Wiren (Ericsson), Drs. Jungwoo Choi (KAIST) and Parker Vascik (MIT) for helpful discussions. This research is partially supported by the Swedish Transport Administration, the Swedish Research Council, CORUS-XUAM project which has received funding from the SESAR Joint Undertaking under the European Union's Horizon 2020 research and innovation programme under grant agreement No 101017682, and AiRMOUR project which has received funding from the EU's Horizon 2020 research and innovation programme under grant agreement 101006601.

REFERENCES

[1] E. Ancel, T. Helsel, and C. M. Heinich, "Ground risk assessment service provider (grasp) development effort as a supplemental data service provider (sdsp) for urban unmanned aircraft system (uas) operations," 2019.

[2] L. C. Barr, R. Newman, E. Ancel, C. M. Belcastro, J. V. Foster, J. Evans, and D. H. Klyde, "Preliminary risk assessment for small unmanned aircraft systems," in *17th AIAA Aviation Technology, Integration, and Operations Conference*, 2017, p. 3272.

[3] E. Denney, G. Pai, and M. Johnson, "Towards a rigorous basis for specific operations risk assessment of uas," in *2018 IEEE/AIAA 37th Digital Avionics Systems Conference (DASC)*. IEEE, 2018, pp. 1–10.

[4] S. Primatesta, A. Rizzo, and A. la Cour-Harbo, "Ground risk map for unmanned aircraft in urban environments," *Journal of Intelligent & Robotic Systems*, vol. 97, no. 3, pp. 489–509, Mar 2020. [Online]. Available: https://doi.org/10.1007/s10846-019-01015-z

[5] A. Washington, R. A. Clothier, and J. Silva, "A review of unmanned aircraft system ground risk models," *Progress in Aerospace Sciences*, vol. 95, pp. 24–44, 2017.

[6] A. Volkert, "Final contingencies & constraints," 2019, CORUS D3.2.

[7] E. T. Dill, R. V. Gilabert, and S. S. Young, "Safeguard," in *2018 IEEE/AIAA 37th Digital Avionics Systems Conference (DASC)*, 2018, pp. 1–8.

[8] J. Breunig, "A risk-based approach for small unmanned aircraft system (suas) airworthiness and safety certification," in *2017 Integrated Communications, Navigation and Surveillance Conference (ICNS)*. IEEE, 2017, pp. 1–28.

[9] K. Dalamagkidis, K. P. Valavanis, and L. A. Piegl, "Current status and future perspectives for unmanned aircraft system operations in the us," *Journal of Intelligent and Robotic Systems*, vol. 52, no. 2, pp. 313–329, 2008.

[10] L. Murzilli, "JARUS guidelines on specific operations risk assessment (SORA)," 2019.

[11] K. Dalamagkidis, K. P. Valavanis, and L. A. Piegl, "On unmanned aircraft systems issues, challenges and operational restrictions preventing integration into the national airspace system," *Progress in Aerospace Sciences*, vol. 44, no. 7-8, pp. 503–519, 2008.

[12] R. A. Clothier and R. A. Walker, "Determination and evaluation of uav safety objectives," in *21st International Unmanned Air Vehicle Systems Conference*, 2006.

[13] D. Haddon and C. Whittaker, "Aircraft airworthiness certification standards for civil uavs," *The Aeronautical Journal*, vol. 107, no. 1068, pp. 79–86, 2003.

[14] R. Weibel and R. J. Hansman, "Safety considerations for operation of different classes of uavs in the nas," in *AIAA 4th Aviation Technology, Integration and Operations (ATIO) Forum*, 2004, p. 6244.

[15] A. V. Shelley, "A model of human harm from a falling unmanned aircraft: implications for uas regulation," *International Journal of Aviation, Aeronautics, and Aerospace*, vol. 3, no. 3, p. 1, 2016.

[16] R. Aalmoes, Y. Cheung, E. Sunil, J. Hoekstra, and F. Bussink, "A conceptual third party risk model for personal and unmanned aerial vehicles," in *2015 International Conference on Unmanned Aircraft Systems (ICUAS)*. IEEE, 2015, pp. 1301–1309.

[17] J. Jung, S. N. D'Souza, M. A. Johnson, A. K. Ishihara, H. C. Modi, B. Nikaido, and H. Hasseeb, "Applying required navigation performance concept for traffic management of small unmanned aircraft systems," in *ICAS*, 2016.

[18] R. Weibel, M. Edwards, and C. Fernandes, "Establishing a risk-based separation standard for unmanned aircraft self separation," in *11th AIAA Aviation Technology, Integration, and Operations (ATIO) Conference, including the AIAA Balloon Systems Conference and 19th AIAA Lighter-Than*, 2011, p. 6921.

[19] J. Holden and N. Goel, "Fast-forwarding to a future of on-demand urban air transportation," 2016.

[20] European Commission, "Implementing regulation (eu) 2019/94," 2019.

[21] R. Storvold, C. Sweatte, P. Ruel, M. Wuennenberg, K. Tarr, M. Raustein, T. Hillesøy, T. Lundgren, and M. Sumich, "Arctic Science RPAS Operator's Handbook," Arctic Monitoring and Assessment Programme (AMAP), Tech. Rep., 2015.

[22] R. E. Weibel and R. J. Hansman, "Safety considerations for operation of unmanned aerial vehicles in the national airspace system," MIT International Center for Air Transportation, Tech. Rep., 2005, Report No. ICAT-2005-1.

[23] EASA, "Acceptable Means of Compliance and Guidance Material to Commission Implementing Regulation (EU) 2019/947– Issue 1," 2019.

[24] G. Sestili, "Q & A on the new common european rules," 2019.

[25] V. Duchamp, L. Sedov, and V. Polishchuk, "Density-adapting layers towards pbn for utm," in *ATM Seminar*, 2019.

[26] T. Prevot, J. Rios, P. Kopardekar, J. E. Robinson III, M. Johnson, and J. Jung, "UAS traffic management (UTM) concept of operations to safely enable low altitude flight operations," in *16th AIAA Aviation Technology, Integration, and Operations Conference*, 2016, p. 3292.

[27] SESAR, "U-space blueprint," 2017. [Online]. Available: https://www.sesarju.eu/sites/default/files/documents/reports/U-space%20Blueprint%20brochure%20final.PDF

[28] FAA, "Operation of small unmanned aircraft systems over people," rIN 2120-AK85. [Online]. Available: https://www.faa.gov/uas/programs_partnerships/DOT_initiatives/media/2120-AK85_NPRM_Operations_of_Small_UAS_Over_People.pdf

[29] S. Young, E. Ancel, A. Moore, E. Dill, C. Quach, J. Foster, K. Darafsheh, K. Smalling, S. Vazquez, and E. Evans, "Architecture and information requirements to assess and predict flight safety risks during highly autonomous urban flight operations," 2020.

[30] P. F. Di Donato and E. M. Atkins, "Evaluating risk to people and property for aircraft emergency landing planning," *Journal of Aerospace Information Systems*, pp. 259–278, 2017.

[31] Eurocontrol, "Final Report of Riga CTR Airspace Assessment," 2019.

[32] P. Butterworth, "GUTMA Annual Conference will be a vital waypoint for the UTM industry," 2018.

[33] R. Dalmau and X. Prats, "Fuel and time savings by flying continuous cruise climbs: Estimating the benefit pools for maximum range operations," *Transportation Research Part D: Transport and Environment*, vol. 35, pp. 62–71, 2015.

[34] T. Andersson Granberg, T. Polishchuk, V. Polishchuk, and C. Schmidt, "Automatic design of aircraft arrival routes with limited turning angle," in *16th Workshop on Algorithmic Approaches for Transportation Modelling, Optimization, and Sys (ATMOS 2016), August 25, 2016, Aarhus, Denmark*, vol. 54, 2016, pp. 9–1.

[35] D. Mitchell, H. Ekstrand, X. Prats, and T. Grönstedt, "An environmental assessment of air traffic speed constraints in the departure phase of flight: A case study at gothenburg landvetter airport, sweden," *Transportation Research Part D: Transport and Environment*, vol. 17, no. 8, pp. 610–618, 2012.

[36] R. Cabell, F. Grosveld, and R. McSwain, "Measured noise from small unmanned aerial vehicles," in *Inter-Noise and Noise-Con Congress and Conference Proceedings*, vol. 252. Institute of Noise Control Engineering, 2016, pp. 345–354.

[37] M. Arntzen, R. Aalmoes, F. Bussink, E. Sunil, and J. Hoekstra, "Noise computation for future urban air traffic systems," in *44th Inter-Noise Congress: Implementing Noise Control Technology*, 2015.

[38] V. Bulusu, V. Polishchuk, and L. Sedov, "Noise estimation for future large-scale small uas operations," in *INTER-NOISE and NOISE-CON Congress and Conference Proceedings*, vol. 254. Institute of Noise Control Engineering, 2017, pp. 864–871.

[39] L. Sedov, V. Polishchuk, and V. Bulusu, "Sampling-based capacity estimation for unmanned traffic management," in *2017 IEEE/AIAA 36th Digital Avionics Systems Conference (DASC)*. IEEE, 2017, pp. 1–10.

[40] G. Scozzaro, D. Delahaye, and A. Vela, "Noise abatement trajectories for a uav delivery fleet," in *SID*, 2019.

[41] E. Rudnick-Cohen, J. W. Herrmann, and S. Azarm, "Risk-based path planning optimization methods for unmanned aerial vehicles over

inhabited areas," *Journal of Computing and Information Science in Engineering*, vol. 16, no. 2, 2016.
[42] Statistics Sweden, "B13: Totalbefolkning på 100x100 m ruta," Swedish University of Agricultural Sciences, http://www.slu.se/en/.
[43] A. Hately et al., "U-space concept of operations vol.2," 2019, https://www.sesarju.eu/node/3411.

BIOGRAPHIES

Dr. **Valentin Polishchuk** is an Associate Professor in the Department of Science and Technology of Linköping University, leading the Academic Excellence in ATM (and UTM) Research group.

Leonid Sedov is a PhD student in Linköping University in Sweden. He works on airspace design, Unmanned Traffic Management, and UAM, in particular on airspace capacity estimation and route optimization.

Dr. **Vishwanath Bulusu** is an Aerospace Research Scientist with Crown Consulting Inc. working at NASA Ames Research Center, under the NAMS contract, currently on the Pathfinding for Airspace with Autonomous Vehicles (PAAV) team.