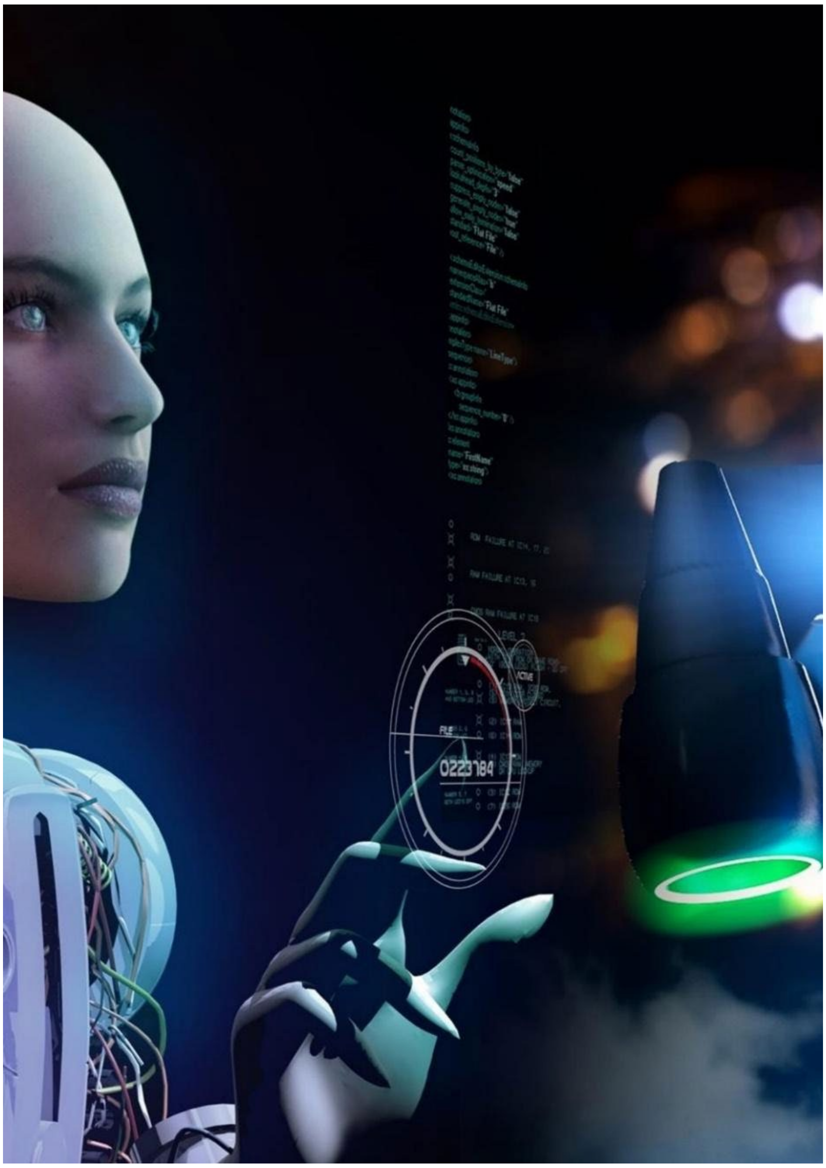
Performance-Based Navigation for Unmanned Traffic Management (PBN4UTM)

Billy Josefsson (LFV)

Leonid Sedov (Linköping University)

Augusti 2023 rev. Jan 2024



Innehåll

1 Sammanfattning

Under projektet PBN4UTM har vi utvecklat flera algoritmer och kvantitativa metoder som bidrar till etableringen av ett prestationsbaserat navigationskoncept (PBN) för framtida hantering av obemannad trafik (UTM). Bland de främsta resultaten ingår algoritmer för sektorisering av luftrum baserat på drönarnas konfliktlösningsprestanda, tillvägagångssätt för att dela upp luftrummet i zoner med liknande operativa höjder, algoritmer för riskmedveten ruttplanering, ett tillvägagångssätt för strategisk konfliktlösning med betalningsmekanismer samt ett tillvägagångssätt för centraliserad storskalig operationsplanering med exempel på leverans av medicinska tester vid pandemier. Det här projektet bygger vidare från resultat som utvecklats inom projektet UTMOK (TrV-finansierat).

Vi implementerade de algoritmer vi designade och utvärderade dem för att visa hur de kan bidra till det framtida UTM-systemet. Vi visade hur våra algoritmer hjälper till att möjliggöra användningen av billigare och mindre kapabla drönare i mindre trafikerade delar av luftrummet med bibehållen säkerhet, samtidigt som höga krav på navigationsförmåga behöver upprätthålls i tätbebyggda stadskärnor. Vårt arbete med höjdmedveten indelning av luftrum tar itu med dilemmat; det gemensamma höjdreferenssystemets (CARS) som diskuteras av EUROCONTROL och EASA i [CARS], genom att visa hur luftrummet kan delas upp så att en enda referenshöjd kan etableras i varje zon. Vi tog fram en kvantitativ metod för att uppskatta och minimera den risk som en drönaroperation medför för människor på marken, vilket är nödvändigt för att digitalisera ruttplanering och riskbedömning. Dessutom resulterade vårt arbete i ett nytt tillvägagångssätt för strategisk konfliktlösning för luftfartyg, som nyttjar betalningssystem för att låta drönaroperatörer påverka prioriteringen av sina operationer. Slutligen presenterade vi hur storskaliga drönaroperationer kan hanteras med ett exempel från massleverans av tester under en pandemi.

Vårt arbete syftar till att hjälpa till att etablera ett säkert och inkluderande luftrum för framtida omfattande operationer med obemannade flygfordon (UAV). Vi följde noggrant utvecklingen av EU-förordningar [EU664,EU665,EU666,EU945,EU947,AMC/GM664,AMC/GM947] samt EUROCONTROL och EASA-diskussioner (t.ex. [CARS]) för att göra vår forskning så tillämplig och aktuell som möjlig.

LFV och LiU samarbetade tätt i detta projekt och samarbetet med forskare från hela världen är betydande och gör vårt arbete relevant inte bara för svenska aktörer och myndigheter utan också för den internationella UTM-gemenskapen. Vår AEAR forskningsgrupp deltog också i relevanta SESAR-finansierade projekt (AiRMOUR, Metropolis 2 och CORUS-XUAM) för att bättre samordna och sprida våra resultat samt få återkoppling från andra stora europeiska aktörer.

2 Summary

During the PBN4UTM project, we have developed several algorithms and quantitative methodologies which serve towards the establishment of performance-based navigation (PBN) concept for future Unmanned Traffic Management (UTM). Among the main results are algorithms for airspace sectorization with respect to the deconfliction performance of drones, approaches for dividing airspace into zones with similar operational altitudes, algorithms for risk-aware route planning, an approach for strategic deconfliction with payment mechanisms, and an approach for centralized large-scale operations planning on an example of medical tests deliveries in case of pandemics. This project builds upon results developed in our earlier (supported by Trafikverket) UTMOK project.

We implemented the algorithms we designed and experimented with them to demonstrate how they can aid the future UTM system. We showed how our algorithms help to facilitate using cheaper and less performant drones in less busy portions of the airspace where it is safe while keeping high requirements on navigational capabilities in the dense city center. Our work on altitude-aware airspace zoning tackles the Common Altitude Reference System (CARS) conundrum discussed by EUROCONTROL and EASA in [CARS] by showing how airspace can be divided so that a single Above Mean Sea Level (AMSL) altitude can be established in every zone. We also presented a quantitative

methodology for estimating and minimizing the risk that a drone operation imposes on people on the ground which aids the complete digitalization of route planning and risk assessment. Next, our work presented a new approach for strategic deconfliction of aircraft which makes use of payment systems to let drone operators influence the priority of their operations. Finally, we discussed how large-scale drone operations can be managed on an example of mass-delivery of tests during a pandemic.

Our work aims to help establish safe and inclusive airspace for future massive Unmanned Aerial Vehicle (UAV) operations. We closely followed the development of EU regulations [EU664,EU665,EU666,EU945,EU947,AMC/GM664,AMC/GM947] as well as EUROCONTROL and EASA discussions (e.g., [CARS]) to make our research as applicable as possible. LFV and LiU worked in tandem on this project and collaborated with researchers from around the world to make our work to be beneficial not only for Swedish authorities but also for the whole international UTM community. Our AEAR research group also took part in relevant SESAR-funded projects (AiRMOUR, Metropolis 2, and CORUS-XUAM) so as to better coordinate and disseminate our results as well as gather input from other major European actors.

3 Results

3.1 Introduction

Unmanned aircraft are not a new technology: The New York Times reported a story of a remotely piloted trans-Atlantic flight back in 1947 [Lev47]. Such one-off endeavors, however, do not scale up, because huge investments and a whole fleet of operation management helpers are required to execute a single flight.

Before long, drones will be flying everywhere, greatly improving the quality of a multitude of services. According to the European drone outlook study [SES16], over 7 million drones are expected in Europe by the year 2025 (of which ~200 thousand will be commercial and governmental drones and approximately one thousand military drones). Federal Aviation Administration (FAA) states in their report [FAA21] that they expect more than 2 million Unmanned Aerial Vehicles (UAVs) by the year 2025 in the United States alone.

We work towards thousands of flights per day in a metropolitan area, managed with little human oversight. We are motivated by the need to establish a guidance and control network for autonomous drones and elaborate on the rules governing the future use of airspace by UAVs.

Indeed, city-scale drone operations pose significant challenges in managing the new type of air traffic and regulating the rules of the game so as not to jeopardize the social potential of the emerging technology while keeping a grasp on it. Addressing this challenge led to the very recent emergence of a whole new discipline — Unmanned Traffic Management (UTM), which is the UAV-focused counterpart of conventional Air Traffic Management (ATM) [COR19, FAA20].

UTM research is still young and remains subject to many uncertainties as to how the actual unmanned traffic will look in the future once massive numbers of commercial drone flights become a reality. There are several Concepts of Operations (ConOpses) developed in different countries that aim to define “the rules of the game”, i.e., how the future UTM will look (see, e.g., [COR19,COR22,FAA20, MET16,PRK+16,Glo17,Air16]) as well as regulations that recently started emerging (see, e.g., [EU664, EU665,EU666,EU945,EU947]). In this project, we have closely followed these developments and treated challenges these ConOpses and regulations mention as an inspiration for our work.

Performance-based navigation (PBN) is a concept initially developed for conventional manned aviation [ICAO08], which proposes establishing different aircraft navigational performance requirements in different airspaces. One of the key benefits of PBN was that it defined requirements in terms of accuracy, integrity, availability, continuity, and functionality (or, in layman’s terms, the equipment performance), as opposed to the previously existing concepts which required specific navigation sensors.

NASA ConOps [PRK+16] and SESAR U-space Blueprint [SES17] later promoted the development of a PBN-like concept for UTM. The central idea is to acknowledge the users' diversity and to provide them with the required level of navigational services and access to airspaces according to their equipment level. Such a PBN concept would make the airspace more accessible to less performant drones, therefore improving the overall airspace capacity utilization.

Metropolitan areas serve as a good example of where this concept could be beneficial, as the city center is expected to experience much higher levels of unmanned traffic, while the less-populated surrounding territories will likely be less congested. Making such less congested areas more accessible for cheaper drones with less precise equipment can improve overall airspace utilization and promote the diversity of airspace users.

Technological implementation of PBN for UTM (and the relevant airspace designs) can be achieved through methods like geofencing (as proposed in U-space U1 foundation services [SES17] and CORUS [COR19] ConOpses) or geovectoring. However, the essential question of how and how often the information about the airspace can be updated remains to be addressed (as also highlighted in [AMC/GM664, GM2 Article 3(4)]. Current practices in conventional ATM, such as updating Aeronautical Information Publication (AIP) through AIRAC cycles, are slow and might not suit the dynamic and uncertain nature of unmanned traffic demand which is easily influenced by various factors like daily patterns and public events.

To address this, the concept of a Software-Defined Airspace (SDA) emerges, which implies frequent (or even on-demand) automatic (or with minimal human intervention) reconfiguration of the airspace design or structure. This information would then need to be communicated to airspace users through appropriate channels, such as, e.g., a geo-awareness service defined in [EU664]. SDA aims to enhance operational efficiency by swiftly adapting the airspace to the changing demand and thereby optimizing capacity utilization.

Our work follows the SDA concept and aims to support the development of autonomous PBN solutions for future UTM. In the remainder of this chapter, we present the research problems we formulated during our work on the project and the solutions we propose along with references to the relevant publications.

3.2 Performance-based airspace sectorization

The work in the earlier Trafikverket-funded UTMOK project (predecessor of PBN4UTM) as well as the SESAR-funded Metropolis project demonstrated the efficiency of the Layered airspace design, where the airspace is divided into a set of sectors (layers) defined by their altitudes (see Figure 1). The idea is that within each layer, drones occupy roughly the same altitude.

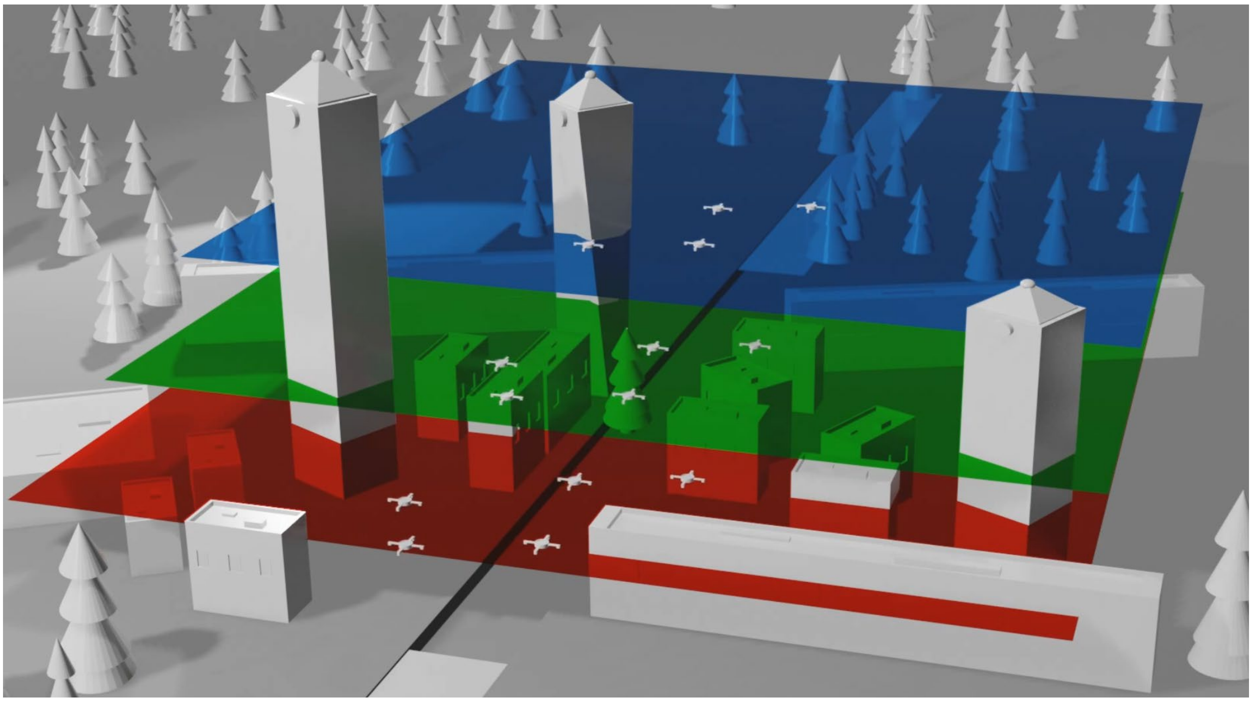


Figure 1 Three layers above a city (red, green, and blue). Drones in two different layers are considered to be separated from each other by altitude.

However, one of the challenges with this approach is that the band of Very Low Level (VLL) altitude airspace where unmanned vehicles are allowed to operate can be rather thin (due to regulations, noise concerns [BSP17], etc.), and the ability of a drone to precisely maintain its altitude depends on its equipage quality. Therefore, having many UTM layers requires UAVs operating in the airspace to have more precise navigational capabilities (as to maintain their altitude reliably and not violate other layers during the operation), which in its turn may raise the price of vehicles. However, it is unlikely that drones demand will be uniform across the whole city: suburban areas often have a more sparse population (a big part of the population also tends to migrate to the city centre for work during office hours) and will likely see fewer operations than the denser city centre. Therefore, it might be of economic and social interest to have different sets of requirements for drones in different parts of a city: while a dense city centre requires more expensive higher-precision drones to operate over it, suburban areas may allow cheaper drones to fly to enable the technology for the wider range of businesses and individuals.

In this project, we have developed a sectorization algorithm that divides a city into several zones where different requirements for drones can be established. Our algorithm is inspired by work on perfect graphs [Gol80] and builds upon earlier work on UTM capacity thresholds [BP17, SPB17]. The novelty of our work in comparison with [BP17, SPB17] is that here we focus on the spatial distribution of the conflicts and suggest how to use it to our advantage.

Our adaptive airspace design follows Mantra 1 “Flexibility where possible, structure only where necessary” of NASA UTM ConOps [PRK+16], and introduces the idea of establishing more layers only where it is required for traffic deconfliction. It also fulfills Mantra 2 “Risk-based approach where geographical needs and use cases determine the airspace performance requirements”, as it separates different UAV classes based on their performance. This is also in line with SESAR’s U-space Blueprint that encourages “To follow a risk-based and performance-driven approach when setting up appropriate requirements...” [SES17]. Therefore, as the altitude band may be thin, then in an airspace with many layers the distance between the layers will be small, implying that only vehicles with good equipment and navigational performance (and, possibly lower noise) will be allowed there — a performance-based requirement.

Our algorithm works as follows. First, it samples drone traffic according to where it is likely to see a drone. Then it builds a conflict graph where, as before, nodes are represented by drones, and two nodes are connected with an edge if the corresponding drones would be in conflict if they were operating on the same layer. In this graph, the algorithm searches for cliques. A clique in graph theory is a subset of nodes of an undirected graph so that every node in that subset is connected with every other node from the subset. Thus, cliques in a conflict graph represent complex conflicts between several drones where each drone conflicts with every other drone (see Figure 2 for an example of a conflict clique). The practical use of cliques here is that when doing graph coloring, all vertices in a clique must have distinct colors, i.e., it is impossible to color the clique of size *n* with fewer than *n* colors. In our UTM setting distinct colors in the conflict graph represent different layers, so if we have a maximum clique of size *n* in the conflict graph, we need at least *n* layers to resolve the conflict by assigning drones to different layers.

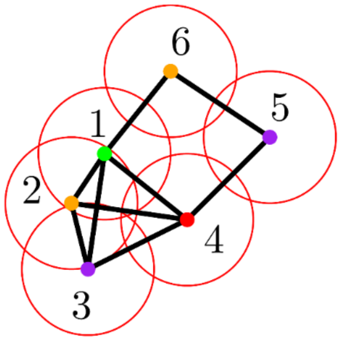


Figure 2 A snapshot of a conflict between six drones. Red circles represent the safety zone around them. Drones 1-4 form a clique of size 4, which means that at least 4 distinct colors are needed to color all nodes in this conflict graph. The picture is reproduced from Figure 1 of [Paper 1].

After finding all large cliques, the algorithm then looks at their geographical distribution and separates the city into different zones based on the observed cliques' sizes. We experimented with several different algorithms for achieving that: convex hull, *k*-hull (a generalization of convex hull that allows leaving some outliers outside), *α*-shape (a generalization of convex hull that allows to produce non-convex polygons that more closely follow the shape of the points), and *k*-order *α*-shape (the generalization of both *k*-hulls and *α*-shapes combining their properties). Figure 3 shows how such layers are defined by using convex hull and *k*-order *α*-shape algorithms.

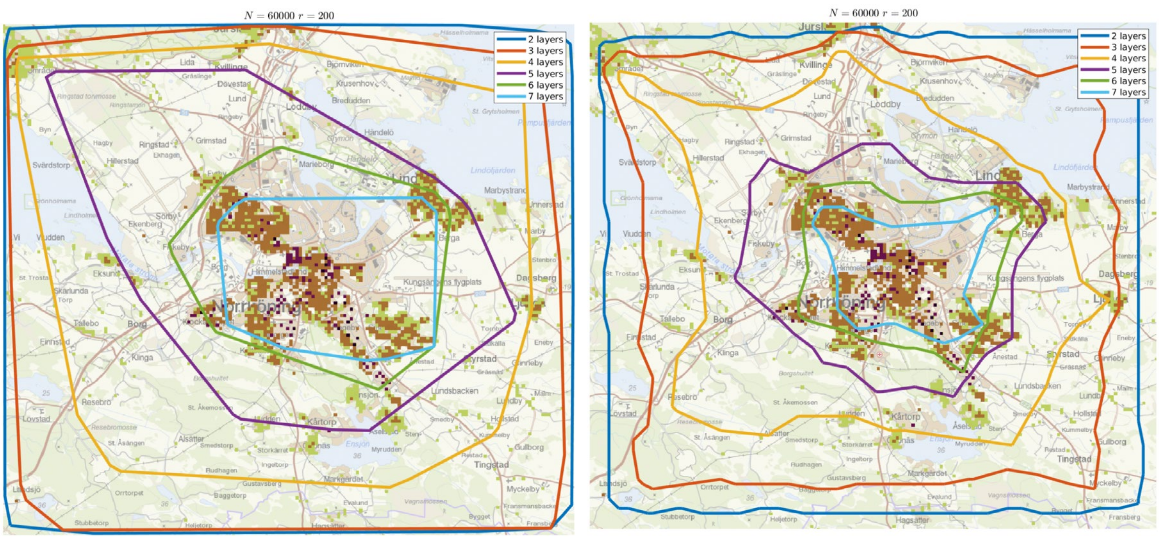


Figure 3 A possible division of the airspace of Norrköping city into zones with different numbers of layers in them. Left: using convex hull for defining geographical zones. Right: using *k*-order *α*-shapes for defining geographical zones. The

heatmap in the background represents the population density with darker colors signifying a denser population. The population data is courtesy of [SLU]. The figure is reproduced from Figures 5 and 12 of [Paper 1].

One of the benefits of having automated algorithms for airspace sectorization like the one developed by us is that it allows authorities to quickly and easily compute new airspace sectorizations, e.g., in case there are significant drone demand changes expected or even to revise airspace sectorizations regularly at scheduled times.

We refer the reader to [Paper 1] for additional details on this study.

3.3 Strategic deconfliction with a payment mechanism for prioritization

After addressing in Section 3.2 how many layers are needed in different regions of the airspace assuming that drones will be deconflicted by being assigned to different layers of the airspace, we now focus on the question of how such strategic deconfliction can be done.

Drones are expected to do numerous short flights, which warrants a higher level of automation for solving conflicts. We consider a scenario where drone operators submit flight plans with a planned trajectory for each flight [PRK+16] and propose an algorithm for strategic deconfliction which works by automatically distributing these flightplans between airspace layers so that every layer is completely conflict-free.

The objective of our algorithm is to minimize the social cost of drone operations. We assume that operating a drone on different layers creates different footprints on both the society and the drone operator due to a variety of reasons, such as, e.g., noise and visual pollution, different levels of difficulty of operating on different altitudes, and so on. Thus, we consider that assigning drone d to layer l would incur some cost c_d^l to society. This cost does not need to be monetary, instead, it can be an arbitrary dimensionless number, representing the comparative preferentiality of different choices. The exact values are specific to the local airspace features and can be chosen by experts (e.g., during the national U-space observatory meetings proposed by EASA in their guidance materials document [AMC/GM664]) based on the quantitative or qualitative assessment of the airspace. In addition to that, the drone operators can also express their own cost w_d^l for operating drone d on layer l (for example, they can assign a higher cost to upper layers if operating at a higher altitude is more difficult for them). Thus the total cost of assigning drone d to layer l is $w_d^l\, c_d^l$ and the goal is to assign all drone operations to layers so that the sum of the costs $w_d^{l(d)}\, c_d^{l(d)}$ (where $l(d)$ is the layer selected for drone d as a result of solving the optimization problem) associated with the chosen layers is minimized.

This problem is closely related to the graph coloring problem discussed in Section 3.2. Again, we build a graph where vertices represent drone missions, and we connect with edges all pairs of vertices where the corresponding drones would be in conflict if they were operating at the same altitude. However, the classical graph coloring problem is concerned only with minimizing the number of colors used (which, in our setting, means minimizing the number of layers used). Instead, we formulate our problem as a Minimum Sum Coloring Problem (MSCP) [JHH17], where every color is assigned a cost (in our UTM setting it is the cost of assigning the drone to the layer), and the objective is to minimize the sum of the chosen colors' costs.

One of the benefits of our formulation is that the same problem formulation can be used for deconfliction based on ground delay in single-layered airspace where all drones operate at the same altitude. In this formulation, the colors now represent time slots, the cost c_d^l is simply the amount of time the drone has to wait on the ground if it was assigned to that slot, and the multiplier w_d^l now represents the cost of time as reported by the drone operator. Then, the same MSCP problem can be solved to obtain the optimal solution, where the total cost of delays (the sum of delay costs, where every delay time $c_d^{l(d)}$ is multiplied by the cost of drone operator's time $w_d^{l(d)}$) is minimized.

However, while our formulations provide drone operators with the ability to influence the priority of their operations, they also open the possibility for drone operators to exploit the system by overestimating their costs w_d^l, since it is beneficial to lie in order to get preferential treatment (either get assigned the best layers or get the lower delays). In order to tackle that issue, we propose a payment mechanism that incentivizes users to truthfully report their respective costs.

The classic result in economics states that any economic mechanism can possess at most three of the four following properties (thus, at least one must be left out): Social Optimality (the resource allocation should maximize the benefit to the society), Incentive Compatible (every participant achieves the best outcome by acting truthfully), Individual Rationality (each user benefits from participating in the market), and Budget Balance (the resource owner's net profit is 0). In our economic mechanism, Social Optimality is achieved by finding the optimal solution to our MSCP as discussed above, and we seek to include the Incentive Compatible property since it would force operators to truthfully report their costs. We chose to forgo the Budget Balance property, and therefore use the socially optimal truthful individually rational Vickrey–Clarke–Groves (VCG) mechanism to determine the payments for the drone operators. In VCG, the payments of user d is the "harm" that the user brings to the society, i.e., the difference p_d between the social welfare of the others in the presence of d and the social welfare they could have gotten if d were not in the society. The full payment for operating the drone d is then $p_d + w_d^{l(d)} c_d^{l(d)}$. Note that all drone operators are paying to the mechanism (i.e., the resource owner only receives money), which means that the Budget Balance is not met. A possible way to introduce budget balancing would be to pay drone operators for surrendering good layers (or time slots) to others; this, however, could create problems, such as incentivizing operators to schedule flights with the only purpose of selling their slots.

Next, we implemented our algorithm and compared its efficiency to a simple First-Come-First-Serve (FCFS) strategy, where drone operations are assigned to the best available layer in the order in which they arrive, that is, no global optimization of all operations is done. We considered a simple scenario with delivery flights in Norrköping Municipality, Sweden. We sampled demand proportionally to the population density, similar to the demand model mentioned in Section 3.2, and generated a series of different 1-hour-long scenarios with the number of drone operations ranging from 100 to 5000. We formulated our MSCP with a payment mechanism as an Integer Program (IP) and solved it using the Gurobi solver [Gurobi]. Figure 4 shows that our MSCP approach outperforms the simple FCFS strategy.

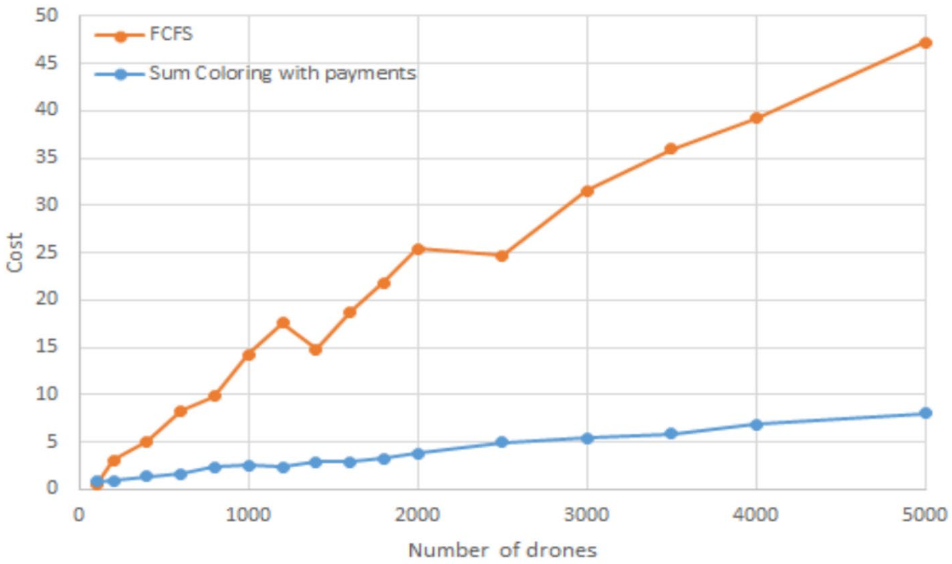


Figure 4 Figure 7 from [Paper 2] demonstrating how our sum-coloring strategic deconfliction algorithm compares to a simple FCFS strategy.

Our MSCP approach is also applicable to the strategic deconfliction of conventional aviation, but since PBN4UTM project focuses on unmanned traffic, in this report we omit the ATM component and refer the interested reader to the original publication [Paper 2] for a more detailed discussion.

3.4 Common altitude reference

When discussing multi-layered airspaces we assumed that drones should follow a certain altitude and mentioned that UAVs will be restricted to a very narrow altitude band. In conventional ATM aircraft are flying at Above Mean Sea Level (AMSL) altitude, which stays the same regardless of the height of the terrain below the aircraft. However, as highlighted, e.g., in Eurocontrol/EASA discussion document [CARS] (and followed upon in SESAR project ICARUS), it will not necessarily be possible for UTM to use AMSL altitude as reference, because drones will be restricted to VLL airspace which in some countries can be as low as 50 meters above the ground, and a city can easily have more than 50 meters difference in the terrain/building elevation within its area. The other alternative is to follow the altitude above terrain level, namely, Above Ground Level (AGL). However, this option is also not necessarily viable as a city can have complex terrain which is hard to follow. Figure 5 illustrates the issues with both approaches.

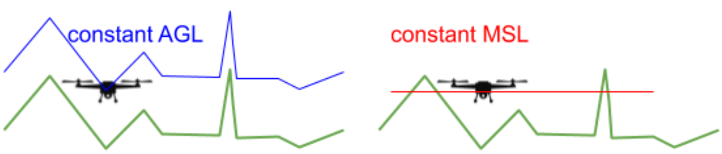


Figure 5 Issues with both AGL (left) and AMSL (right). The picture is reproduced from Figure 1 of [Paper 3].

To accommodate for this problem, SESAR Exploratory Research [SES19, Area 2.7.2] has called for the development of a common altitude reference system (CARS) for UTM and manned aviation. Eurocontrol's UAS ATM airspace assessment [Hul18] also states that guidelines for CARS are necessary.

As a solution to this problem, we propose dividing a city into several zones so that in every zone drones can follow the same AMSL altitude without risk of colliding with buildings or terrain. We formulated two different optimization problems to divide the city into such similar-altitude zones and we developed algorithms that solve them.

3.4.1 Terrain partitioning

The first optimization problem deals with rural terrains and small cities, where there are no (or a small number of) very tall buildings. Given a discrete terrain elevation map (a grid of points with known elevation), we want to partition the terrain into rectangular zones so that in any zone the difference between the minimum and the maximum terrain elevation is smaller than a certain threshold that represents the upper bound of the airspace reserved for unmanned operations. Figure 6 presents the idea on a 1d example. While this threshold can deviate between different countries and their specific UTM regulations, as an example in this work we choose 60 meters to be the threshold. The objective is then to minimize the number of zones (to have as few altitude changes and as least complex airspace as possible).

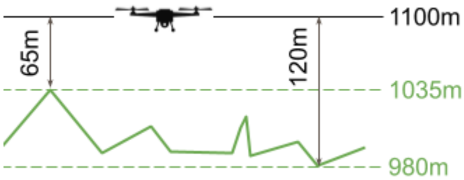


Figure 6 1d example of the terrain partitioning problem. The difference in altitude between the lowest and the highest points on the terrain is 55m, which means that the UAV keeps an altitude of at least 65 meters above the highest point while always staying less than 120m from the point with the lowest elevation. The picture is reproduced from Figure 2 of [Paper 3].

To solve this problem we developed two approaches. The first approach uses an IP formulation to optimally divide the airspace into the minimum possible number of zones so that the elevation difference constraints are upheld. While the IP formulation produces an optimal solution, it takes a long time to compute and requires a lot of memory even on terrains with relatively small resolutions with the problem complexity growing exponentially (we experimented with up to 60x60 points terrains, with higher resolutions being tricky to tackle even on supercomputer clusters). Therefore, we also developed a heuristic algorithm that does not necessarily output the optimal solution but often outputs good solutions, and it does it very quickly which allows working with higher-resolution terrains (we experimented with up to 2000x2000 terrains — more than a 30-fold increase in the resolution in every dimension).

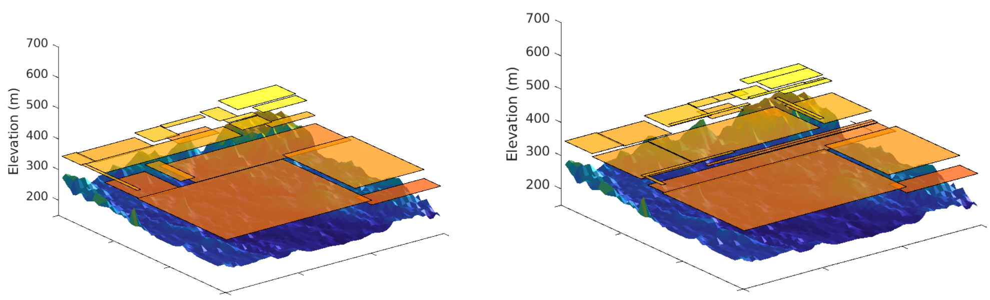


Figure 7 Partitioning of the terrain around Eksjö Municipality. Left: IP formulation. Right: heuristic algorithm. The data is courtesy of [Lantmäteriet]. The graphs are reproduced from Figures 4 and 11 of [Paper 3].

We implemented both the IP formulation and the heuristic algorithm and applied them to the terrain around Eksjö Municipality in Sweden. Figure 7 shows the resulting zoning of the terrain after applying both approaches. We can see that while the heuristic algorithm produced a slightly worse solution than the optimal solution obtained by solving the IP formulation, it still did a considerably good job at the benefit of having a solution almost instantaneously.

To benchmark the performance of our heuristic and how well it performs compared to optimal solutions, we tested it on multiple random terrains. We observed that our heuristic usually produces solutions with the number of zones at most 2 times the optimal number.

3.4.2 Urban partitioning

The second optimization problem deals with a heavily urban setting, where there are tall buildings of height greater than the allowed threshold (with height more than 60 meters above the ground). In this scenario, it is impractical to apply the previous optimization problem, because the drones are no longer guided by the terrain elevation but by the need to avoid tall buildings. In addition to that, applying the previous terrain partitioning model would result in every tall building having a separate zone above it.

We formulate the optimization problem for the urban scenario as follows. First, we “fatten” all buildings by 50 meters which represents the requirement for drones to keep a distance of at least 50 meters from buildings [EU947] (the number can easily be adjusted to any other value to fit other regulations). Then we suggest partitioning the city into a fixed number *K* of rectangular zones (every rectangle is elevated at the altitude above the highest building height it covers) so that the volume under these zones is minimized. Basically, we want to ”pack” the city into *K* boxes of minimum total volume.

The reasoning for packing the city into the minimum total volume boxes is twofold. First, cities with a big number of tall buildings tend to have airports located near them, and this approach allows to “steal” as little airspace from conventional aviation (which often requires a good portion of the urban airspace close to the ground; see [VCBP20] for a quantitative study of urban airspace availability) as possible. Second, flying more than 50 meters away from a building while being at an altitude matching the height of that building would technically lead to the violation of the height restrictions. Our approach helps to minimize the number of such violations.

As with the terrain partitioning version, we have developed an IP formulation that solves the problem to optimality and a heuristic algorithm that produces a feasible solution quickly, but without any quality guarantees. We experimented on elevation data from New York City in an area around the Empire State Building. Figure 8 shows the 3D profile before and after fattening the buildings. The complexity of the IP formulation of the problem has only allowed us to solve problems with building height data of resolutions at most 45x45 points (Figure 9 left). The heuristic algorithm produced a solution for the same resolution almost instantaneously (Figure 9 right).

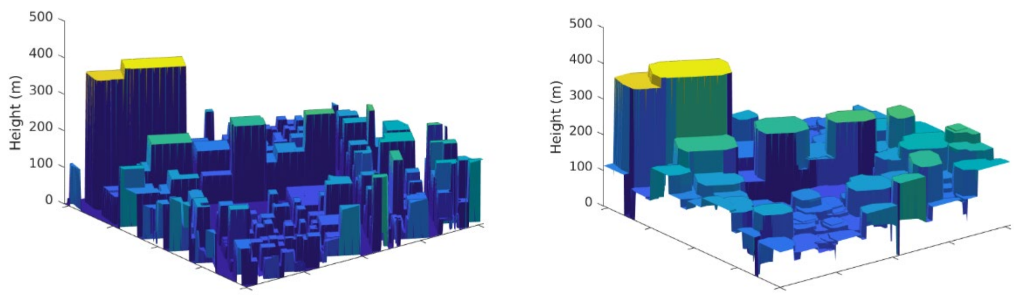


Figure 8 Buildings profile in New York City before “fattening” (left) and after “fattening” buildings by 50 meters (right). The building height data is courtesy of [NYC14]. The visualizations are reproduced from Figure 7 of [Paper 3].

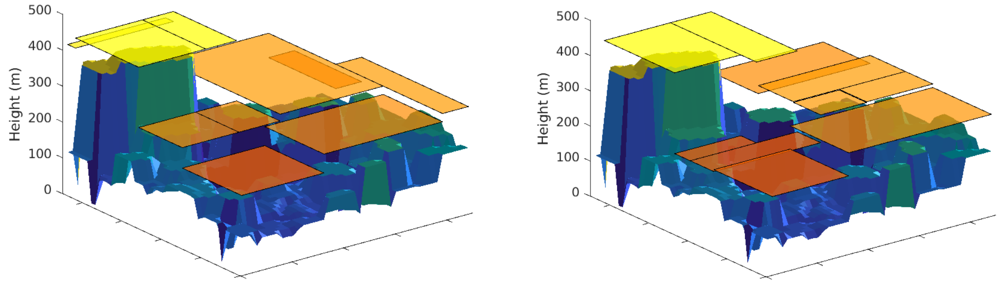


Figure 9 The results of solving the Urban Partitioning problem using IP formulation (left) and heuristic algorithm (right). The graphs are reproduced from Figures 8 and 12 of [Paper 3].

We experimented with our heuristic algorithm further and we tested it on 30 different building profiles from New York City and compared results with the optimal solutions. Our heuristic algorithm is based on simulated annealing technique [KGV83] and relies on randomization to produce solutions. Nevertheless, we have observed that despite the random nature, the heuristic tends to produce solutions with less than a 10% increase in the objective function value. Using the heuristic we have been able to work with terrains of resolution greater than 2000x2000 points which is a drastic increase in the height data resolution.

To keep this report focused, we omit the technical description of the algorithms we developed and we refer the reader to [Paper 3] for a formal definition of both IP formulations and heuristic algorithms and for a deeper discussion of this study.

3.5 High-Fidelity Risk Modeling

The next topic which we studied in this project deals with the safety aspect of UAV operations.

Contrary to the conventional ATM, where aircraft carry people on board and therefore the research focuses on the first-party risks to the passengers and the cabin crew, in UTM drones do not carry any passengers, so there is only the third-party risk of injuring people on the ground (financial losses from losing a drone are overshadowed by the safety of people).

Risk assessment is at the core of UAVs operations, so ground risks from UAVs operations have been widely studied [AHH19, BNA+17, DPJ18, PRlCH20, WCS17]. One of the most promising developments is the Specific Operations Risk Assessment (SORA) methodology developed by JARUS [SORA]. SORA equips operators with a qualitative tool for drone operations risk assessment. The qualitative user-oriented nature of the methodology means that it consists of simple questions with straightforward answers, which essentially allows users to complete the risk assessment by only using a piece of paper and a pen, as no computing or data processing is required. SORA quickly gains popularity in Europe: there are thousands of performed SORA assessments and the number continues to grow.

However, the qualitative nature of the SORA methodology also holds certain disadvantages, in particular in the ground risk assessment category. The results of performing SORA on a flightplan are Ground Risk Class (GRC) and Air Risk Class (ARC), and once the flightplan is fulfilling the desired ARC or GRC, there is no motivation for the operator to make any changes or improvements to the flightplan. The ground risk assessment in SORA is particularly vulnerable to this issue because it only looks at the highest population overflown, regardless of what are the mean or median values. For example, from SORA's point of view, all the routes presented in Figure 10 are going to have the same GRC because they all have the starting and the endpoints located in areas with dense population, and it does not matter whether the middle part of the route goes above unpopulated areas or the same dense urban areas, as it does not change the risk class.

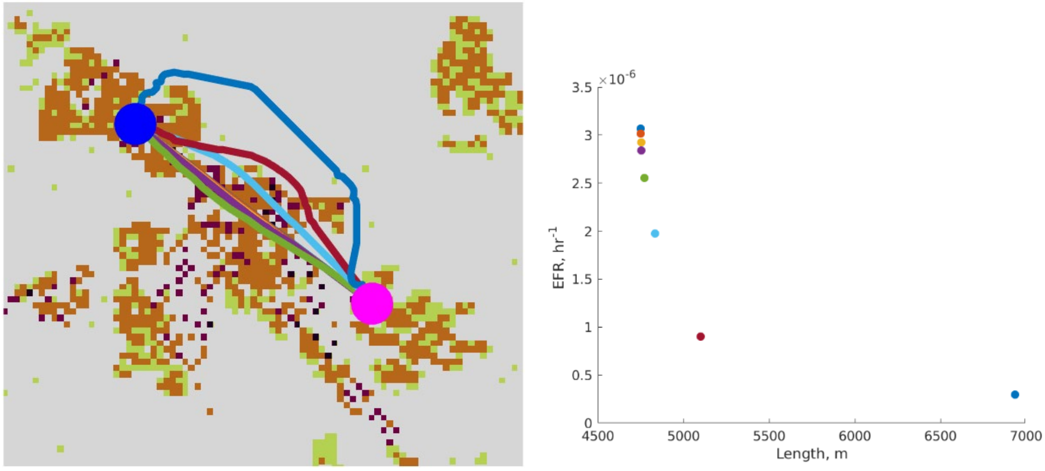


Figure 10 A set of different routes between the same start and end points (left) and a graph of their risks and lengths (right). All these routes have the same risk class in the SORA methodology because their origin and destination are located in densely populated areas. The figures are reproduced from Figure 6 of [SPM+21].

CORUS ConOps [COR19] also noted how Air Risk Class (ARC) discourages attempts to decrease GRC: "drone operations in ARC-c or ARC-d offer little motivation for an operator to reduce the GRC, as the SAIL [Specific Assurance Integrity Level] stays almost unaffected." [COR19, Section 2.2.2.c]. In this work, we emphasize how GRC discourages attempts to decrease the exposed to risk population.

To avoid this shortcoming we suggest using a quantitative approach, where ground risk created by a flightplan is computed by estimating the number of people exposed to the risk from the operation. In our experiments, we used an estimation approach similar to the one mentioned in [PRICH20] (but slightly simplified; however any other model can easily be substituted instead of ours): we assume that for a given route segment the drone can fall anywhere within distance r of this route segment with uniform probability and compute the exposed population by simply computing the sum of people in all points of the population density map covered by a rectangle with width $2r$.

It is important to acknowledge that there is a trade-off between the route length and the ground risk imposed by the route. Operators may not always wish to take the safest route if it is too long for them. Based on these assumptions, we developed an algorithm that for two given points computes several routes. To account for the trade-off between the length and the risk, each of the produced routes has a minimum length for a given number of affected people and affects the fewest number of people for its length (i.e., we output so-called Pareto-optimal solutions).

Our algorithm takes a discrete population density map, origin point, and destination point as an input, and produces a set of Pareto optimal routes. It works by building a complete graph where the population density map points are nodes, and an edge between each pair of points has two labels that represent the distance between the points and the number of people exposed to risk when flying between the points.

One approach to solving the problem would be to compute all Pareto-optimal routes, by using a Dijkstra-like algorithm, where for every edge instead of keeping one label, the algorithm should maintain a table of labels (the best lengths for every exposed population value). However, this approach has several disadvantages, such as the fact that the problem is NP-hard, and therefore would take a lot of time to compute, and that it would often result in an excessive number of Pareto-optimal routes.

Therefore, we also suggest another approach which is to find only the lower hull of the Pareto frontier (i.e., only a subset of the Pareto optimal routes, see Figure 11). Our algorithm does that by finding the minimum-length path (P_1) and the minimum-risk path (P_{end}) from the origin to the destination in this graph using a shortest path algorithm (in our experiments we used Dijkstra, but any other algorithm would work) using as the edge weights the length and the exposed population values respectively. Our algorithm then computes the slope β between P_1 and P_{end}, and looks for a shortest path in a graph where all edge weights are set to $l + \beta * p$, where l is the length and p is the exposed population along the edge. If the newly found route is the same as one of the P_1 or P_{end}, then the algorithm stops. Otherwise, the algorithm recurses on both sides of the newly found route and checks whether or not there is another extreme point (if yes, then it recurses further until no new points are found).

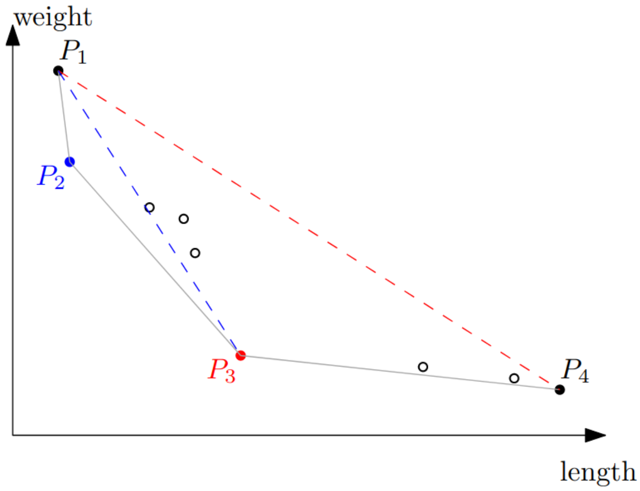
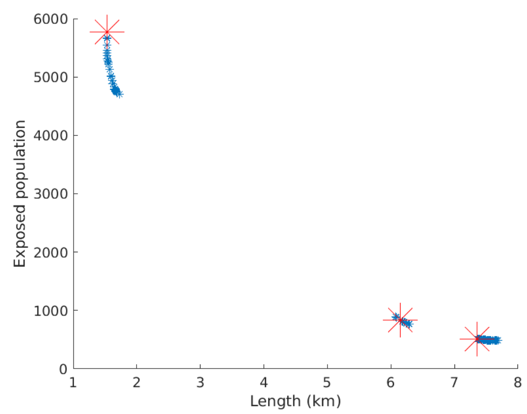


Figure 11 Circles give example lengths and weights of Pareto-optimal paths. P_1, ..., P_4 are the paths on the lower hull of the Pareto frontier (the hull is shown with the thin line). P_3 is the extreme point of the frontier between the extreme Pareto optimal solutions P_1 and P_4. To find the other points on the lower hull, we recurse on both sides of P_3: on the right, no new solutions are found; on the left, a new Pareto optimal path P_2 is discovered as the extreme point between P_1 and P_3. The graph is reproduced from Figure 3 of [Paper 4].

We tested our algorithm on an example of Norrköping Municipality in Sweden. We selected two points as our origin and destination and run the algorithm to find the routes on the lower hull of the Pareto frontier. Figure 12 shows the paths that resulted from our algorithm, while Figure 13 shows three selected routes (marked with asterisks on Figure 12).



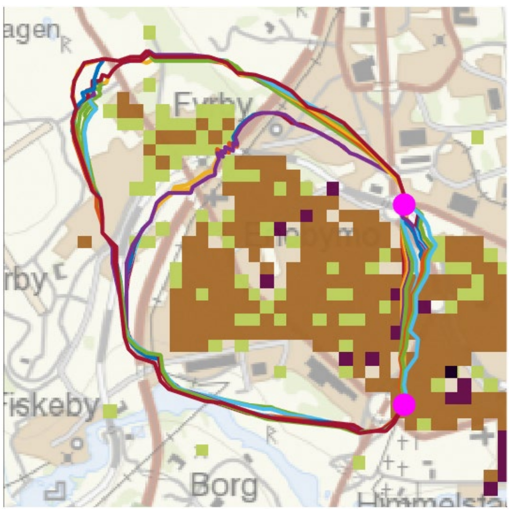


Figure 12 Routes on the lower hull of the Pareto frontier. Left: the lower hull of the Pareto frontier (red asterisks mark the routes shown on Figure 7). Right: the routes. The graphs are reproduced from Figure 6 of [Paper 4].

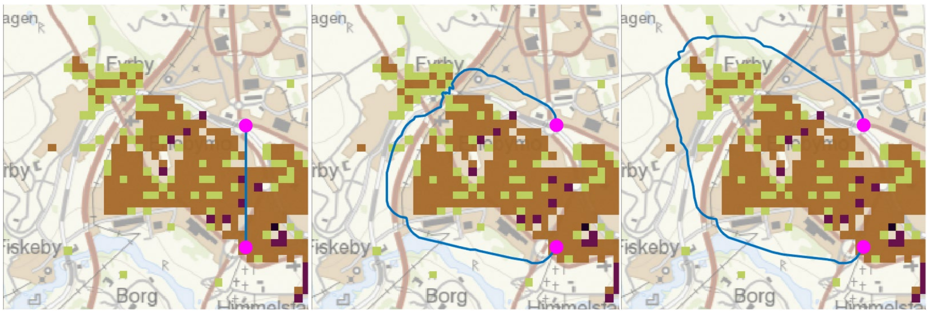


Figure 13 Selected routes from Figure 12 (marked with red asterisks there). The length increases and the risk decreases from left to right. The visualizations are reproduced from Figure 7 of [Paper 4].

In addition to that, we have developed an interactive web prototype that allows users to play with their own routes and look at how the ground risk changes. This prototype does not compute the routes by itself (as it is a rather computationally expensive task) but instead makes it possible to see how changing a route changes the associated ground risks. We invite the reader to check it out at https://undefiened.github.io/ground_risk/

Results from this section are discussed in more detail in our publications [Paper 3] and [SPM+21].

3.6 Centralized planning of large-scale delivery operations

Last, we discuss how drone delivery operations can be planned on a large scale using an example of home delivery of tests during a pandemic.

Late 2019 and early 2020 became characterized by the global COVID-19 virus outbreak, which was declared a public health emergency of international concern by the World Health Organization [CV20]. While the majority of people experience only comparatively mild symptoms of COVID-19 (if any at all, given the high reported number of asymptomatic cases), the few people, particularly the elderly and those with preexisting underlying health issues, were experiencing severe symptoms requiring hospitalization and use of life support machines. As the disease spread faster and faster, affecting millions of households and causing hundreds of thousands of deaths, health authorities in many countries became increasingly worried about the shortage of hospital spaces suited to support people with more difficult cases of the disease.

SIR (**S**usceptible, **I**nfectious, **R**ecovered) model [KM1927] is one of the classical compartmental models in epidemiology which describes how the population moves between three different compartments as the spread of the disease progresses in time. In the SIR model, the population transitions from the Susceptible compartment (healthy people who can get sick by contacting the disease) to the Infectious compartment (sick people who are spreading the disease) as they get sick, and then to the Recovered compartment (people who recovered from the disease and developed immunity) as they get healthy again. One of the interesting observations that this model makes, is that first the disease spreads quicker as more people become infectious (since more people spread the disease simultaneously), but then, after the peak, the spread speed starts declining as more people gain immunity after transitioning into the Recovered compartment. One of the main parameters which governs the speed with which a disease spreads in the SIR model is called the basic reproduction number (R_0) — the expected number of cases directly generated by one case in a population where all individuals are susceptible to infection. The lower values of the basic reproduction number mean that the disease spreads slower, and, hence, the peak number of simultaneously infectious people is lower and achieved later.

These observations lead health authorities around the world to pursue the so-called “flatten the curve” strategy during the COVID-19 pandemic, which intended to reduce R_0 and thus flatten the disease spread curve, hoping to reduce the maximum number of people who need help simultaneously and to gain time to increase the healthcare capacity.

The measures taken for reducing R_0 during the COVID-19 pandemic ranged between countries and often included increased hygienic requirements (such as face mask mandates, single-use hand gloves mandates, informational campaigns teaching the proper techniques of hands washing, stands with hand-cleaning solutions in public places, and so on), limiting the number of social contacts (lockdowns, restricting public events, mandating working from home, restricting non-essential businesses activity, and many others), and isolating the infectious people (increased testing for the disease, mandating staying at home when experiencing symptoms of COVID-19, etc.). As restrictive measures are easing around the world, the balance between the harm and the efficiency of such measures remains a widely debated topic to this day.

Our work in [Paper 5] follows the "flatten the curve" strategy by suggesting regular home delivery of tests via drones to the entire population of the affected region. Such proactive screening aims to capture people who are only developing the first symptoms (as viral diseases tend to spread even before the infected person starts exhibiting first symptoms) as well as the asymptomatic cases who may remain unaware that they were spreading the disease during its entire course [BYW+20,FLN+20,HSX+20,LPC+20,NKM+20,WNB20]. The general idea of this approach is simple: tests are regularly delivered to the people who do the testing by themselves (either collect the test specimen and send it back to the laboratory, or, if testing technology allows, read the response by themselves) and, if test results return positive, self-quarantine until they recover.

In order to tackle the problem of scheduling such large-scale delivery of tests to the entire population we identified and proposed solutions for two complementary subproblems: the first problem is to quantify the effect of regular testing on the disease spread curve and the second problem is to plan the regular delivery of tests using a limited park of drones.

In [Paper 5] we quantify the effect of regular testing of the entire population on the disease spread by extending the classical SIR model with an additional Quarantined compartment — we dubbed our model an SIQR model. In our model, people transition from the Infectious compartment into the Quarantined compartment at a rate that depends on the frequency of regular testing. People in the Quarantined compartment are then assumed to self-isolate until they become healthy, and, therefore, transition to the Recovered compartment without infecting other people. Figure 14 shows an example of the disease spread evolution in our SIQR model without any testing and with regular testing every D = 8 days.

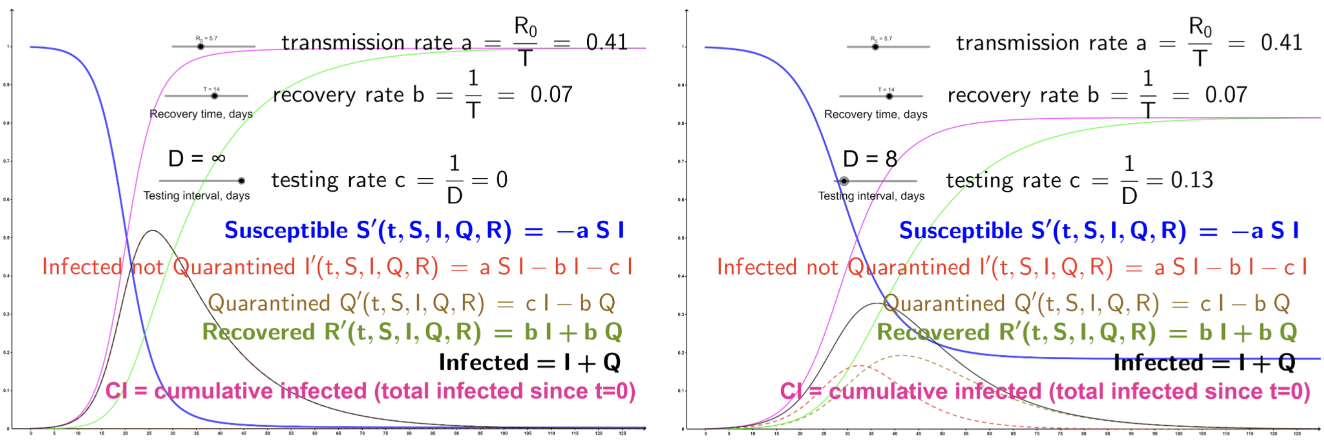


Figure 14 Figure 3 from [Paper 5] shows the curves for a quite high value of R_0 = 5.7. Left: SIR curves without any testing (obtained by setting the testing interval to infinity). Right: SIQR with a testing interval of 8 days. These pictures were obtained using our online applet, we invite the curious reader to play with different values of R_0 and testing rate at http://tiny.cc/SIQR to see how higher testing frequency flattens the curve.

Next, we discuss how to plan such large-scale operations. We assume a model, where all tests are delivered in small batches from a single facility (e.g., a hospital or a warehouse) to drop sites at or close to homes (i.e., one per apartment complex or a cluster of single-family houses). The tests are to be delivered by a given park of drones with some capacity *C*. We also assume that drones can carry several small batches of tests and make multiple stops along one flight as long as the drone capacity C is not exceeded. We tackle this problem in two steps. The first step is to find a set of tours — individual flights that start from the hospital, visit some drop sites, drop test batches (the number of which is limited by the capacity constraint), and return back to the hospital — covering the entire population. Minimizing the total length of such tours, needed to cover all people in the area, is known as the Capacitated Vehicle Routing Problem (C-VRP). The second step, after finding a set of tours, is to distribute them between the available drones. We call a set of tours assigned to a single drone a route. The objective now is to deliver all tests as quickly as possible, i.e., to reduce the length of the longest route, and, therefore, the time when the last person will receive a test. This problem is known as the Minimum Makespan Scheduling problem (or Multiprocessor Scheduling where drones are "processors").

We implemented both problems and experimented with them on the example of Norrköping Municipality in Sweden. The first step with C-VRP was implemented and solved using Google OR-tools [ORtools], and the second step (the Minimum Makespan Scheduling problem) was formulated as an Integer Program and solved using the Gurobi solver [Gurobi] on Tetralith supercomputer cluster [Tetralith] provided by the Swedish National Infrastructure for Computing (SNIC). We run our experiments on different scenarios with drone capacity ranging from 20 tests to 1000 tests and varying numbers of drones being available. Figure 15 shows how the number of days needed to deliver tests to the entire population depends on the capacity of drones and their number (please note, that the graph assumes that the tests need to be collected afterward and delivered back to the laboratory, that is, in our simulations all tours are flown twice — first time to deliver tests, and then to collect them). This number is essentially the testing interval from our SIQR model. Figure 16 shows an example of a route consisting of several tours from our experiments.

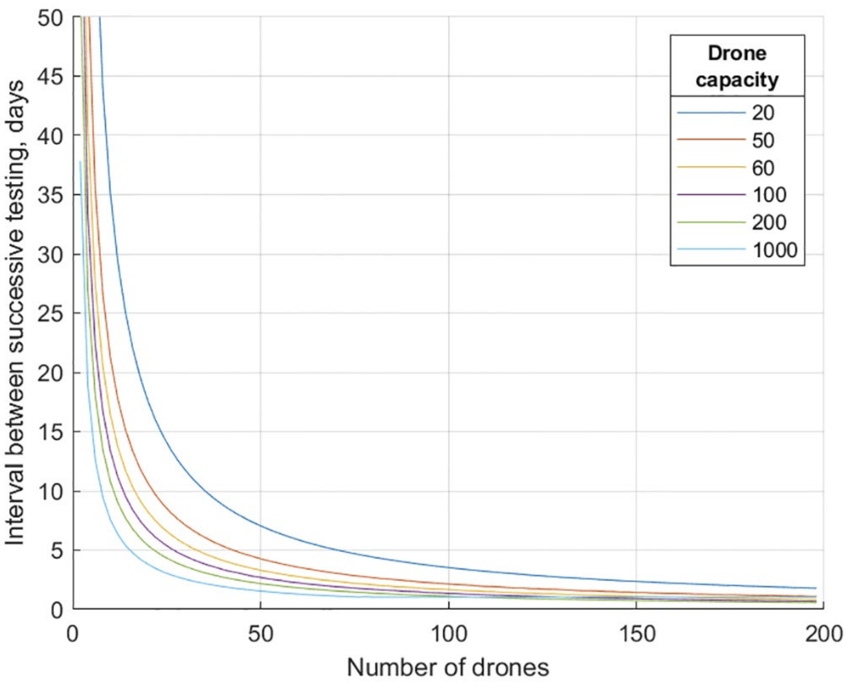


Figure 15 The number of days needed to collect tests from the whole city population as the function of the number of used drones (of different capacities). Reproduced from Figure 1 of [Paper 5].

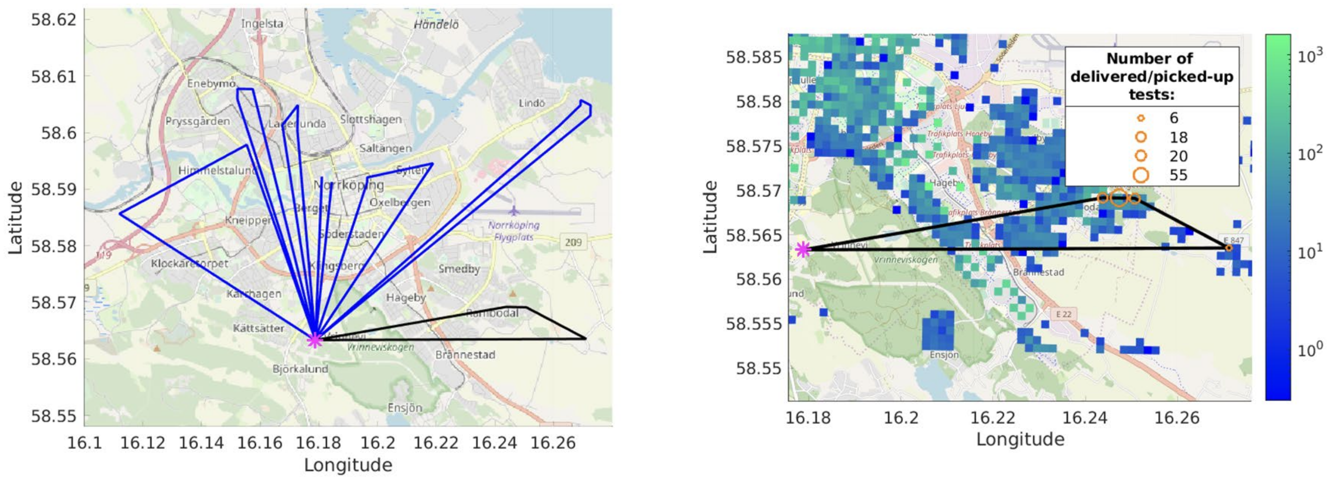


Figure 16 Figure 2 from [Paper 5] showing an example of drone routes in Norrköping Municipality. Left: An example route of a drone with a capacity of 100; one of the tours in the route is shown in black. Pink asterisk is the hospital. Right: A zoom-in on the black tour: the orange circles depict the test delivery/pickup locations; circle size is proportional to the number of tests delivered/picked-up at the location (99 tests are delivered on this tour). The underlying heatmap is the population density.

Separating the problem of planning drone deliveries and the problem of quantifying their influence on the disease spread curve makes it possible to apply our algorithms in both cases when drones are scarce and when they are an abundant resource. If the number of available drones is fixed, then our approaches allow authorities to estimate how often testing of the entire population can be done given the drone fleet, and thus apply our SIQR model to estimate how the disease spread curve will be affected. Otherwise, if authorities simply need to decide how many drones they need to buy, then the SIQR model can be applied first to estimate how often the testing needs to be done in order to achieve an acceptable effect on the disease spread curve, and the route planning algorithms can be run next to determine how many drones are needed to deliver all tests within the given number of days.

We refer an interested reader for more technical discussion to the original publication [Paper 5] and to the source code of the implementation we used to plot Figures 15 and 16 available at https://github.com/undefiened/corona_drones.

3.7 Conclusions

Development of the UTM system is a complex task that requires addressing a multitude of different challenges. During this project, we have developed algorithms and demonstrated different approaches that cover some of the missing components crucial for the development of the future UTM system. We used a modular approach, which means that every algorithm that we developed can be used either in isolation to solve the individual problem, or, if needed, it can be combined with other approaches to work in synergy.

We presented algorithms for sectorizing a Layered airspace into zones with different numbers of layers (or different complexity of conflicts) so that cheaper drones still can operate in less congested areas while only expensive drones with high navigational precision are allowed in the city center. We proposed an algorithm for strategic deconfliction of unmanned flights that allows drone operators to influence the priority of their flight and uses a payment mechanism for incentivizing them to do so truthfully. We developed algorithms for supporting the Layered airspace design by dividing the airspace into zones with constant AMSL altitudes. Our work explored how to compute and minimize the number of uninvolved people exposed to the risk from the unmanned operation. And, finally, we discussed how large-scale delivery operations can be planned on an example of tests delivery during a pandemic.

Our work was motivated by the PBN concept from conventional aviation and aimed to make the future unmanned airspace more inclusive for different types of drones and operations by promoting establishing high technical requirements only where it is necessary.

4 Deliverables

The results of this project are presented in more detail in the following scientific publications:

[Paper 1] V. Duchamp, L. Sedov, V. Polishchuk, “Density-Adapting Layers towards PBN for UTM“, ATM Seminar’19, 2019
[Paper 2] V. Duchamp, B. Josefsson, T. Polishchuk, V. Polishchuk, R. Sáez, R. Wiren “Air Traffic Deconfliction Using Sum Coloring” DASC’19
[Paper 3] L. Sedov, V. Polishchuk, V. Acuna, “Altitude zoning for UTM”, SESAR Innovation Days (SID 2020), 2020
[Paper 4] L. Sedov, V. Polishchuk, V. Bulusu, “Ground risk vs. Efficiency in Urban Drone Operations”, ATM Seminar'21, 2021
[Paper 5] L. Sedov, A. Krasnochub, V. Polishchuk “Modeling quarantine during epidemics and mass-testing using drones”, PLoS One

In addition to dissemination of the results within the international scientific community, we enjoyed a wide public outreach promoting the project among the general audience and industry professionals:

1. Our PhD students who worked on this project have won EUROCONTROL Masterclass Vertiports Challenge (https://www.facebook.com/eurocontrol/posts/pfbid0ivokGQA6sGaGnUQjLCs9WjJw25XrzmqnrzyQhsxVtnRX8gzje7cdJh6xJwNqSfsbl)
2. A submission based on the results of this project was shortlisted at TRA VISIONS 2022 Young Researcher Competition (see the submission by Leonid Sedov at https://www.travisions.eu/TRAVisions/young_researcher_results_2022/)
3. SESAR Young Scientist Award 3rd place in 2019 (Leonid Sedov)
4. DailyMail news (https://www.dailymail.co.uk/sciencetech/article-8254327/Coronavirus-tests-delivered-peoples-homes-using-DRONES-study-suggests.html)
5. SVT news (https://www.svt.se/nyheter/lokalt/ost/dronare-forskning)
6. in CANSO news (collaboration with LFV)
7. video (18min), CANSO *Safety&Ops Forum'20* (held virtually)
8. Video (10min) on our UTM/AAM research (@CORUS-XUAM SESAR VLD)
9. CORUS-XUAM demo

In addition to that, based on the project results, a doctoral dissertation was written and is scheduled to be defended on December 8th, 2023. The presentation is scheduled to start at 10:15 in room K2 in Kåkenhus building of campus Norrköping of Linköping University, Norrköping, Sweden; the detailed invitation with the abstract will be available at https://liu.se/nyheter/kalender.

All papers, presentations and videos produced within the project are openly available on the web. Links to the webpages of the AEAR group members containing the respective materials are available at https://sites.google.com/view/aeargroup/home.

5 Next steps

During the project, based on our findings and discussions with our industrial and academic colleagues (both local and international), we have identified several key directions for future research.

When working on [Paper 4] we noticed that the topic of quantitative risk assessment of unmanned operations receives significant attention. It is crucial for stakeholders to plan for the future, when the expected high levels of unmanned traffic will make manual risk assessment economically unattractive, so the entire risk assessment will need to be done autonomously. Therefore, more research on HFRM approaches for ground and air risk assessment, as well as algorithms for their reduction, are warranted. In addition to that, it is also of interest how such approaches can complement qualitative approaches like SORA in the current early stages of UTM, allowing the initial adoption of the concept.

We will continue our work on HFRM approaches in the Quantitative Risk Assessment (QRA) project, which is essentially a spin-off from PBN4UTM, where we will discuss how quantitative risk assessment solutions can complement SORA and develop new algorithms for risk reduction.

Another alley of future research revolves around the concept of reasonable time to act (RTTA) mentioned in CORUS and CORUS-XUAM ConOpses [COR19, COR22]. This concept is based on the conundrum of when a flightplan should be strategically deconflicted (e.g., by using the algorithm presented in Section 3.3) after it is submitted by the drone operator:

- Deconflicting too early decreases the airspace capacity — the plans may be canceled/modified, implying that the airspace reserved for the original flightplan may be wasted.
- On the other hand, deconflicting close to the actual flight discourages early planning and may disappoint the operator whose plan is rejected by the USSP (because the plan conflicts with an earlier submitted flightplan) at the time when the operation is about to start.

So far very little has been done by the research community to quantitatively determine how different RTTA values affect the airspace capacity and fairness, so in the sister RTTA project, we will explore the trade-offs between different values of RTTA. We will also determine how a PBN-like concept can be established for RTTA, where RTTA may vary for different users in different airspaces, and develop new algorithms for balancing the tradeoffs between airspace capacity and fairness caused by different choices of RTTA values.

One aspect which was not in focus of this project is location of vertiports — one of the main infrastructural considerations in Urban Air Mobility (UAM). In a follow-up project we would like to develop tools to optimize vertiport locations, involving stakeholders from the emerging UAM sector. Last but not least, our work tacitly assumed that drones may fly anywhere over a city. Delineating the available metropolitan airspace (and designing route networks connecting the vertiports through the available airspace) is a subject of a forthcoming research proposal from the LFV—LiU team.

6 Kontakter på LFV Research & Innovation och Digitalisering

Mr Billy Josefsson, Program Manager Automation & Human Performance
billy.josefsson@lfv.se,
+46 708 192331

Gustaf Fylkner, Operativ Specialist , LFV FoID
Gustaf.fylkner@lfv.se
+46737535445

7 References

[AHH19] Ersin Ancel, Tyler Helsel, and Christina 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.

[Air16] Airservices. Management of Remotely Piloted Aircraft Systems (RPAS) in ATM operations, 2016.

[AMC/GM664] EASA. Acceptable Means of Compliance and Guidance Material to Regulation (EU) 2021/664 on a regulatory framework for the U-space, 2019.

[AMC/GM947] EASA. Acceptable Means of Compliance and Guidance Material to Commission Implementing Regulation (EU) 2019/947– Issue 1, 2019.

[BNA+17] Lawrence C Barr, Richard Newman, Ersin Ancel, Christine M Belcastro, John V Foster, Joni Evans, and David H Klyde. Preliminary risk assessment for small unmanned aircraft systems. In 17th AIAA Aviation Technology, Integration, and Operations Conference, page 3272, 2017.

[BP17] V. Bulusu and V. Polishchuk. A threshold based airspace capacity estimation method for UAS traffic management. In IEEE Systems Conference, 2017.

[BSP17] V. Bulusu, L. Sedov, and V. Polishchuk. Noise estimation for future large scale small UAS operations. In NOISECON, 2017.

[BYW+20] Bai Y, Yao L, Wei T, Tian F, Jin DY, Chen L, et al. Presumed asymptomatic carrier transmission of COVID-19. Jama. 2020;323(14):1406–1407.

[CARS] EASA/EUROCONTROL. UAS ATM CARS (Common Altitude Reference System), 2018. Discussion Document.

[COR19] CORUS. U-space Concept of Operations. https://www.sesarju.eu/node/3411, 2019.

[COR22] CORUS-XUAM consortium. U-space ConOps (Edition 3.10). https://corus-xuam.eu/wp-content/uploads/2022/11/CORUS-XUAM-D4.1-delivered_3.10.pdf

[CV20] D. Cucinotta and M. Vanelli. “WHO Declares COVID-19 a Pandemic”. en. In: Acta Biomed 91.1 (Mar. 2020), pp. 157–160.

[DPJ18] Ewen Denney, Ganesh Pai, and Marcus Johnson. Towards a rigorous basis for specific operations risk assessment of UAS. In 2018 IEEE/AIAA 37th Digital Avionics Systems Conference (DASC), pages 1–10. IEEE, 2018.

[EU664] European Commission. Commission Implementing Regulation (EU) 2021/664 of 22 April 2021 on a regulatory framework for the U-space, 2021. Official Journal of the European Union.

[EU665] European Commission. Commission Implementing Regulation (EU) 2021/665 of 22 April 2021 amending Implementing Regulation (EU) 2017/373 as regards requirements for providers of air traffic management/air navigation services and other air traffic management network functions in the U-space airspace designated in controlled airspace, 2021. Official Journal of the European Union.

[EU666] European Commission. Commission Implementing Regulation (EU) 2021/666 of 22 April 2021 amending Regulation (EU) No 923/2012 as regards requirements for manned aviation operating in U-space airspace, 2021. Official Journal of the European Union.

[EU945] European Commission. Commission Delegated Regulation (EU) 2019/945 of 12 March 2019 on unmanned aircraft systems and on third-country operators of unmanned aircraft systems, 2019. Official Journal of the European Union.

[EU947] European Commission. Commission Implementing Regulation (EU) 2019/947 of 24 May 2019 on the rules and procedures for the operation of unmanned aircraft, 2019. Official Journal of the European Union.

[FAA20] Federal Aviation Administration. Federal Aviation Administration - Concept of Operations v2.0.
https://www.faa.gov/uas/research_development/traffic_management/media/UTM_ConOps_v2.pdf, 2020.

[FAA21] Federal Aviation Administration. FAA Aerospace Forecast. Fiscal Years 2021-2041, 2021.

[FLN+20] Ferguson N, Laydon D, Nedjati-Gilani G, Imai N, Ainslie K, Baguelin M, et al. Report 9: Impact of non-pharmaceutical interventions (NPIs) to reduce COVID19 mortality and healthcare demand. MRC Centre for Global Infectious Disease Analysis; 2020.

[Glo17] Global UTM Association. UAS Traffic Management Architecture, 2017.

[Gol80] Martin Charles Golumbic. Algorithmic Graph Theory and Perfect Graphs. Academic Press, Courant Institute of Mathematical Science, New York University, 1980.

[Gurobi] Gurobi Optimization, LLC. Gurobi Optimizer Reference Manual; 2020.
http://www.gurobi.com

[HSX+20] Hu Z, Song C, Xu C, Jin G, Chen Y, Xu X, et al. Clinical characteristics of 24 asymptomatic infections with COVID-19 screened among close contacts in Nanjing, China. Science China Life Sciences. 2020; p. 1–6.

[Hul18] Peter Hullah. UAS ATM Airspace Assessment, 2018.

[ICAO08] International Civil Aviation Organization. Performance-based Navigation (PBN) Manual. 2008.

[JHH17] Y. Jin, J.-P. Hamiez, and J.-K. Hao, “Algorithms for the minimum sum coloring problem: a review,” Artificial Intelligence Review, vol. 47, no. 3, pp. 367–394, 2017.

[KGV83] S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi. Optimization by simulated annealing. Science, 220(4598):671–680, 1983.

[KM1927] Kermack W.O., McKendrick A.G. A contribution to the mathematical theory of epidemics. Proceedings of the royal society of London Series A, Containing papers of a mathematical and physical character. 1927;115(772):700–721.

[Lantmäteriet] Lantmäteriet, “Laserdata NH 2019 (laz),” 2019. [Online]. Available: https://www.lantmateriet.se/

[Lev47] Anthony Leviero. Robot-piloted plane makes safe crossing of atlantic. In The New York Times, 1947.

[LPC+20] Li R, Pei S, Chen B, Song Y, Zhang T, Yang W, et al. Substantial undocumented infection facilitates the rapid dissemination of novel coronavirus (SARS-CoV-2). Science. 2020;368(6490):489–493. pmid:32179701

[MET16] METI. Roadmap for Application and Technology Development of UAVs in Japan, 2016.

[NKM+20] Nishiura H, Kobayashi T, Miyama T, Suzuki A, Jung S, Hayashi K, et al. Estimation of the asymptomatic ratio of novel coronavirus infections (COVID-19). medRxiv. 2020;.

[NYC14] NYC DoITT, “NYC 3-D Building Model,” 2014. [Online].

[ORtools] Perron L, Furnon V. OR-Tools. Available from: https://developers.google.com/optimization/

[Paper 1] V. Duchamp, L. Sedov, V. Polishchuk, “Density-Adapting Layers towards PBN for UTM“, ATM Seminar’19, 2019

[Paper 2] V. Duchamp, B. Josefsson, T. Polishchuk, V. Polishchuk, R. Sáez, R. Wiren “Air Traffic Deconfliction Using Sum Coloring” DASC’19

[Paper 3] L. Sedov, V. Polishchuk, V. Acuna, “Altitude zoning for UTM”, SESAR Innovation Days (SID 2020), 2020

[Paper 4] L. Sedov, V. Polishchuk, V. Bulusu, “Ground risk vs. Efficiency in Urban Drone Operations”, ATM Seminar'21, 2021

[Paper 5] L. Sedov, A. Krasnochub, V. Polishchuk “Modeling quarantine during epidemics and mass-testing using drones”, PLoS One

[PRK+16] Thomas Prevot, Joseph Rios, Parimal Kopardekar, John E Robinson III, Marcus Johnson, and Jaewoo Jung. UAS traffic management (UTM) concept of operations to safely enable low altitude flight operations. In 16th AIAA Aviation Technology, Integration, and Operations Conference, page 3292, 2016.

[PRlCH20] Stefano Primatesta, Alessandro Rizzo, and Anders la Cour-Harbo. Ground risk map for unmanned aircraft in urban environments. Journal of Intelligent & Robotic Systems, 97(3):489–509, 2020.

[SES16] SESAR. European Drones Outlook Study, 2016.

[SES17] SESAR. U-space blueprint, 2017.

[SES19] SESAR. Exploratory research call, 2019.

[SLU] Swedish University of Agricultural Sciences. http://www.slu.se/en/.

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

[SPB17] 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), pages 1–10, 9 2017.

[SPM+21] Leonid Sedov, Valentin Polishchuk, Thibault Maury, Maria Ulloa, and Darya Lykova. Qualitative and quantitative risk assessment of urban airspace operations. In SESAR Innovation Days 2021, 2021.

[Tetralith] Tetralith server, NSC, Linkoping University;. https://www.nsc.liu.se/systems/tetralith/

[VCBP20] Parker D Vascik, Jungwoo Cho, Vishwanath Bulusu, and Valentin Polishchuk. Geometric approach towards airspace assessment for emerging operations. Journal of Air Transportation, pages 1–10, 2020. Special issue on ATM Seminar 2019.

[WCS17] Achim Washington, Reece A Clothier, and Jose Silva. A review of unmanned aircraft system ground risk models. Progress in Aerospace Sciences, 95:24–44, 2

[WNB20] Wang CJ, Ng CY, Brook RH. Response to COVID-19 in Taiwan: big data analytics, new technology, and proactive testing. Jama. 2020;323(14):1341–1342.