Master Thesis

Master of Science in Geospatial Technologies

3D flight route optimization for air-taxis in urban areas with Evolutionary Algorithms

Author
Moritz Hildemann

Supervisors
Jun. Prof. Judith A. Verstegen[1]
Dr. Carlos Granell[2]
Assoc. Prof. Mauro Castelli[3]

Westfälische Wilhelms-Universität Münster (WWU), Institute for Geoinformatics (ifgi), Münster, Germany[1]
Universitat Jaume I (UJI), Castellón, Dept. Lenguajes y Sistemas Informaticos (LSI), Castellón, Spain[2]
Universidade Nova de Lisboa (UNL), NOVA - Information Management School (NOVA IMS), Lisboa, Portugal[3]

February 24, 2020

Declaration of Academic Integrity

I hereby confirm that this thesis “3D flight route optimization for air-taxis in urban areas with Evolutionary Algorithms” is solely my own work and that I have used no sources or aids other than the ones stated. All passages in my thesis for which other sources, including electronic media, have been used, be it direct quotes or content references, have been acknowledged as such and the sources cited.

24/02/2020, Münster M. Hildemann
Date, Location Signature

I agree to have my thesis checked in order to rule out potential similarities with other works and to have my thesis stored in a database for this purpose.

24/02/2020, Münster M. Hildemann
Date, Location Signature

Acknowledgements

I wish to express my sincere appreciation to my supervisors. First of all I want to thank Professor Judith Verstegen, who has mainly guided me through the thesis. Her advices helped not only to improve the quality of this work, but also allowed to improve my own competencies in scientific working. I want to thank my co-supervisors Carlos Granell and Mauro Castelli, who were so kind to support me with their advices. Especially the input regarding the optimization techniques and the reproducibility of this work helped a lot. Furthermore, I wish to acknowledge the support of my parents during my whole master degree. The master program would not have been possible without their help. I want to express my gratitude to my family and friends who supported me in the thesis process. Especially my father and Neele helped a lot in polishing this thesis. Also, I want to point out my friend Carlos, with whom I started the air taxi navigation project last year and who contributed many parts of constructing the restricted airspace.

Contents

List of Figures

3D flight route optimization for air-taxis in urban areas with Evolutionary Algorithms

Moritz Hildemann

Abstract Electric aviation is being developed as a new mode of transportation for the urban areas of the future. This requires an urban air space management that considers these aircraft and restricts the vehicles' flight routes from passing no-fly areas. Flight routes need to be determined that avoid the no-fly areas and are also optimally planned in regard to minimize the flight time, energy consumption and added noise. The no-fly areas and the flight routes can be best modelled as three-dimensional geographical objects. The problem of finding a good flight route that suits all three criteria is hard and requires an optimization technique. Yet, no study exists for optimizing 3D-routes that are represented as geographical objects while avoiding three-dimensional restricted areas. The research gap is overcome by optimizing the 3D-routes with the multi-criteria optimization technique called Non-dominated Sorting Genetic Algorithm (II). We applied the optimization on the study area of Manhattan (New York City) and for two representatives of different electrical aircraft, the Lilium Jet and the Ehang 184. Special procedures are proposed in the optimization process to incorporate the chosen geographical representations. We included a seeding procedure for initializing the first flight routes, repair methods for invalid flight routes and a mutation technique that relocates points along a sine curve. The resulting flight routes are compromise solutions for the criteria flight time, energy emission and added noise. Compared to a least distance path, the optimized flight routes were improved for all three objectives. The lowest observed improvement was a noise reduction by 36% for the Ehang 184. The highest improvement was an energy consumption reduction by 90% for the Lilium Jet. The proposed representation caused high computation times, which lead to other limitations, e.g. a missing uncertainty analysis. With the proposed methods, we achieved to optimize 3D-routes with multiple objectives and constraints. A reproducibility self-assessment[1] resulted in 2, 2, 2, 2, 1 (input data, preprocessing, methods, computational environment, results).

Moritz Hildemann
ifgi, Heisenbergstraße 2 Münster, e-mail: jhildema@uni-muenster.de

[1] https://osf.io/j97zp/

1 Introduction

The traffic in today's Megacities is congested and the commuting times in these cities are high. Therefore, a new green mode of transport has been developed during the past years to relief the transport system: Electrical Vertical Take-Off and Landing vehicles (eVTOL). Due to the advantages of fast travelling in urban areas, the expected potential demand in the highly competitive air taxi market [1] only in the United States is 80.000 passengers per day [2]. A reduced number traffic accidents, traffic decongestion, decreased pollution and also reduced strain on existing public transport networks are expected benefits with the market introduction of eVTOL vehicles [2].

However, the integration of the air taxis into the urban air space is not possible without designing an air traffic management (ATM) which addresses the flight law, other airspace users and also the city population. The airspace needs to be segmented into fly and no-fly zones, depending on the location and the time [3]. This air space division ensures that safety boundaries are protecting permanent locations of special safety concern like hospitals and also occasional no-fly zones in severe weather conditions or during public assemblages of the city population.

Having such a framework for airspace segmentation, optimal flight routes are needed to be computed for the air taxis. But the objectives for the optimal route differs with the viewer's perception: for the commuter for example, the commuting time and price might be the most relevant criteria [4]. For the operator of the air taxi, the commuting time and the energy cost for the flight route might be the crucial criteria. Whereas for the city population, the noise pollution and discomfort produced by the aircraft might be the most important criteria [5]. A method is demanded which incorporates the search for optimal flight paths in with multiple constraints and multiple objectives.

Geographical Information Systems (GIS) enable to compute the time and attribute dependent restricted airspace with dynamic geofences, if the necessary data is available [6]. Yet, no 3D-routing exists which searches solutions for multiple criteria in an 3D environment with can be adapted to different temporal conditions. This adaptability is necessary for incorporating temporal no-fly areas as one example. One approach in GIS for finding the optimal paths including a height component involves cost surface planes. The height is included in the cost value at each cell of the cost surface. With this cost surface, the least cost path can be computed for each criterion to travel from a starting to the destination point following along each cell of the cost surface [7]. The outcome is a two-dimensional line with each node extracted to the height value of the cost at the cell. The drawback of this method is, that some objects planes, submarines or even space shuttles can move in space without being bound to a surface. Following the surface therefore limits the possibility of finding a better solution in the three-dimensional space as indicated in Fig. 1. Therefore, this method is not appropriate for the given problem.

Another possible GIS tool for 3D navigation is 3D-Routing along 3D-Networks. This solves the combinatorial problem to travel from a starting to a destination node of an existing network [8]. On street networks, the possible solutions are limited by the existing streets. The equivalent to this in the 3D-space is pipe routing [9]. The main problem is finding the optimal pipe routes in order to introduce these into a 3D-Network. Finding the optimal route is a hard problem to solve, even for

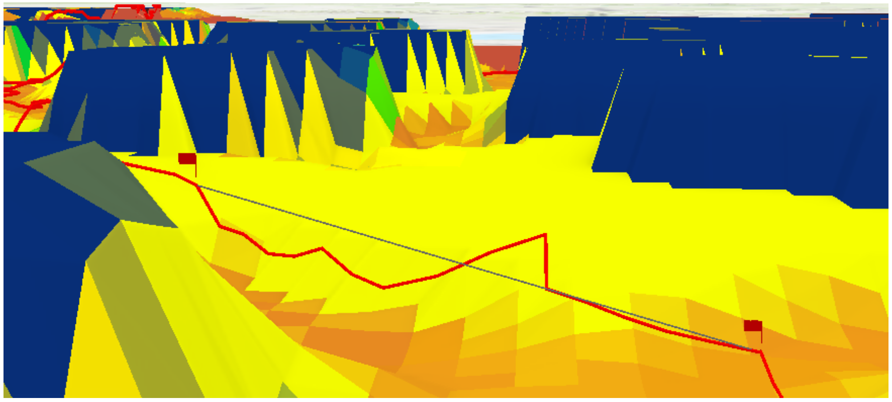

Fig. 1: Example illustration of least cost path on cost surface (red) and shortest 3D path (grey) between two points

one criterion. Hence, a continuous optimization is the appropriate method to find a good solution in a limited time frame [10]. Evolutionary Algorithms (EA), which include Genetic Programming, Evolutionary Strategies, Evolutionary Programming and Genetic Algorithms, yield search techniques for continuous optimization [10]. Sandurkar and Chen [9] solved a 3D routing problem in a Computer-aided design (CAD) environment with the use of Genetic Programming, in order to find the shortest path while avoiding the created 3D CAD objects. The drawback of using CAD objects instead if GIS objects is the non-variable environment, that can be adapted to suite special conditions. One special and time-dependent condition can for example a bigger distance to primary schools in during the mornings or an additional no-fly area during a public event. The CAD objects, in this case the 3D geofences, are not extrudable based on their properties saved in a database, they are just drawn objects [11]. Therefore they are not useful for solving the problem where the environment can change temporally. This means that the so called search space with all possible solutions to solve the problem is not adjustable [12]. Peng et al. [13] also solved the three-dimensional routing problem with multiple constraints and two objectives with Genetic Algorithms, but they also did not consider a adjustable search space.

This work aims to fill the stated research gap to optimize 3D routes with multiple constraints and multiple objectives with a variable search space using a hybrid of GIS and the optimization technique Evolutionary Algorithms. The generalizable method will be applied in a study case to optimize the flight routes of urban air taxis in New York City. The routes will be evaluated for three different criteria, which are specific for the 3D routing in the air taxi transportation: The flight time, the energy emission and the noise pollution. Therefore, one objective for the study is to derive objective functions for the three objectives that assesses the quality for each generated solution. In the case of the flight taxis, it is the computation of the total flight time in minutes, the total energy consumption in kilowatt per hour and the additional noise pollution in decibels for every valid solution. A second objective is to find appropriate representation for the evolutionary algorithms and to choose genetic operators which will suit best for solving the problem in the GIS environment. For finding optimal solutions near the true pareto optimal front for all objectives, the Non-dominated Sorting Genetic Algorithm II is used [14]. The results of the algorithm are a set of non-dominated solutions forming a pareto front. The decision maker can select a solution from this pareto front by defining an importance for each objective.

2 Methods

This section describes the process of bridging the identified research gap. At first, the general idea of evolutionary algorithms is explained and how the idea can be applied to optimize 3D routes (Sec. 2.1). Secondly, the study case for optimizing 3D flight routes as one 3D routing application is laid out (Sec. 2.2). Information about the required input data, e.g. the restricted air space or the minimal flight height. The study case was specifically applied for two types of aircraft in the study area of Manhattan, New York. Thirdly, the representation and evaluation of the flight routes is explained in detail (Sec. 2.3). It describes, which representation was used for the flight routes and how invalid flight routes are identified. Furthermore, the evaluation of the valid flight routes is laid out for all three objectives (Sec. 2.3.3). Fourthly, the multi-objective optimization with evolutionary algorithms is explained with the used Non-dominated sorting algorithm and the used evolutionary operators (Sec. 2.4). Lastly, two selected illustration techniques for depicting the non-dominated solutions are presented (Sec. 2.5).

For an overview over the whole proposed workflow for the 3D route optimization is illustrated in Fig. 2. The structure of this section follows the steps of the illustrated workflow.

2.1 The general idea of Evolutionary Algorithms and the problem-specific use case

The principle of evolutionary algorithms mimics the biological evolution of cells, organs, individuals and populations [10]. The Darwinian theory of natural selection proposes that organisms, that are fitter than other organisms, have a higher probability to survive and passing on the genes in reproduction processes. The idea behind the evolutionary algorithm is to find the individuals, which have the highest fitness, in other words are the best solution for a given problem. These individuals have a higher probability of being selected for reproduction. Evolutionary algorithms involve the idea by producing a population consisting of several individuals in a defined environment [10]. The term "Evolutionary Algorithms" is used as a family name for search techniques mimicking the biological evolution with selection, crossover and mutation processes [10]. The selection process consists of selecting parent solutions based on their fitness for reproduction of children. The crossover process determines,

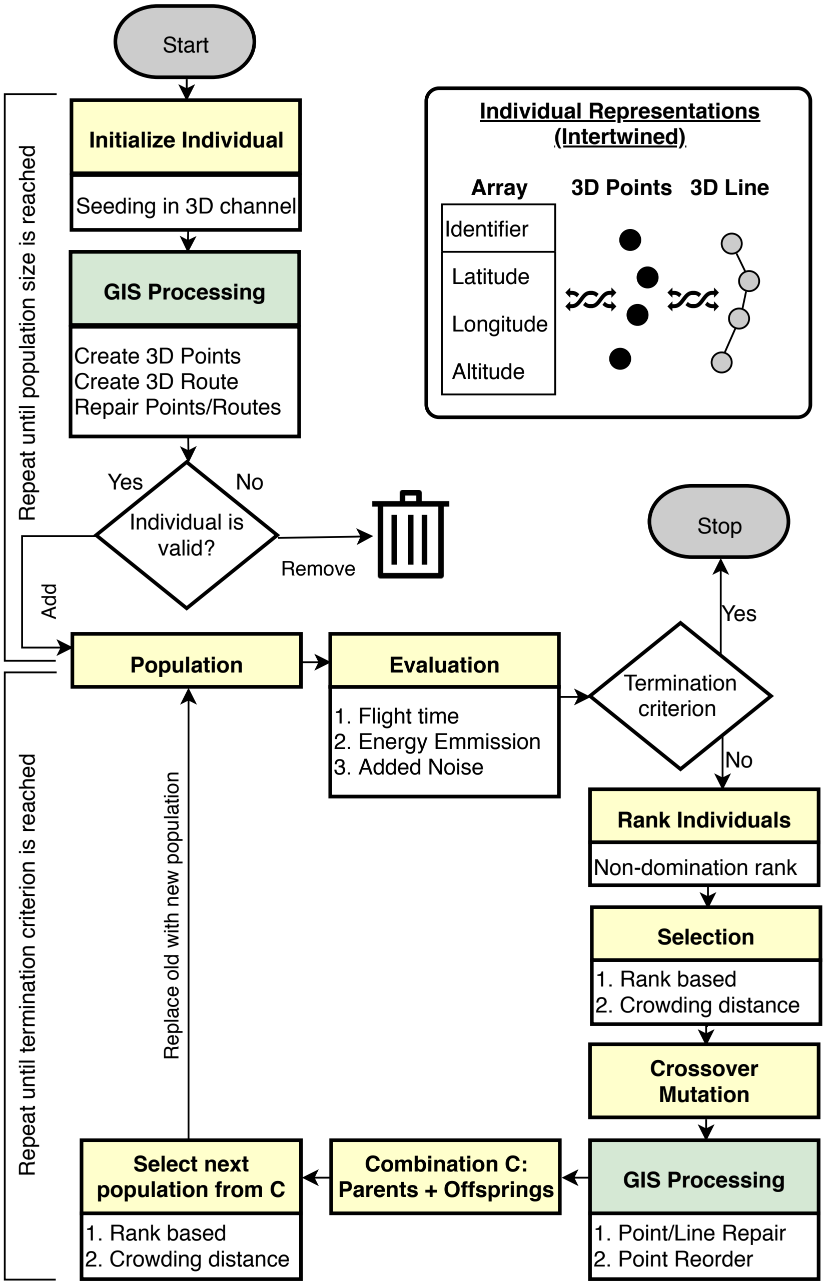

Fig. 2: Flowchart of the used strategy of Evolutionary Computation

how the elements of the parents are recombined to produce a child. The mutation process incorporates characteristics that are not inherited by the parents by randomly perturbing elements [15]. The process of producing generations goes on until the termination criterion is attained. The termination criterion can be, that the maximum number of produced generations is achieved. Other termination criteria can be, that the quality of the individuals are sufficiently good or the variety of the population is low [10].

Each individual of a population is a possible solution for a problem, in our case the problem of finding a 3D flight route between a starting and an ending point. The population is therefore an agglomeration of different solutions for a problem. The idea of computational algorithms is used to produce optimal flight routes for the objectives shortest flight time, lowest energy consumption and minimal added noise level.

The single steps of the workflow are described in detail Sec. 2.3 and in Sec. 2.4, but the structure is laid out shortly in the following and visualized in Fig. 2. One individual is one 3D flight route. All individuals together form the current population. Each individual is validated and evaluated. An individual is invalid, if it violates one of the defined constraints, in our case a constraint violation can be an interference with a restricted flight area. If the individual is valid and has a higher fitness than the other individuals, the probability is high that the individual is selected for reproduction and mutation. The result and crossover and mutation are new solutions, the so-called children. The process of selection, crossover and mutation is repeated until there are as many children as parents. If this loop is finished, a new generation is produced. Depending on the evolutionary computation strategy, the new generated population can exist only of the children solutions, or the new population also incorporates solutions of the parent generation. In the selected strategy NSGA II, the population incorporates solutions of the parent generation [14]. The selected termination criterion is a maximum number of generations, for a better comparison of the convergence behaviour with different input parameters to the optimization.

2.2 Case study description, flight path initialization and implementation

The urban area of New York (Fig. 3) is an often used study area for air taxi marketing [16] and air taxi market studies [2], as the traffic is congested and the city yields one of the highest market potential [2]. Moreover, most of the air space is probable to be regulated with barriers. The barriers can be categorized into

- Restricted air space due to flight regulations: Areas around airports with a distance of 25.000 feet (7620 meters), minimal flight height of 500 feet (152.4 meters) feet above buildings [17], maximal flight height due to the plane air traffic of 700 feet (213.36 meters) [18]
- Protected airspace to consider the possible negative side effects to the citizens [3]. Flying above these specified land uses can lead to annoyance or discomfort: Examples are schools, graveyards, recreational areas

The restricted and protected areas are modelled as 3D-geofences with Geographical Information Systems. The vertical and horizontal minimum distance to be maintained by aircraft depends on the specific land use, which is retrieved from Open

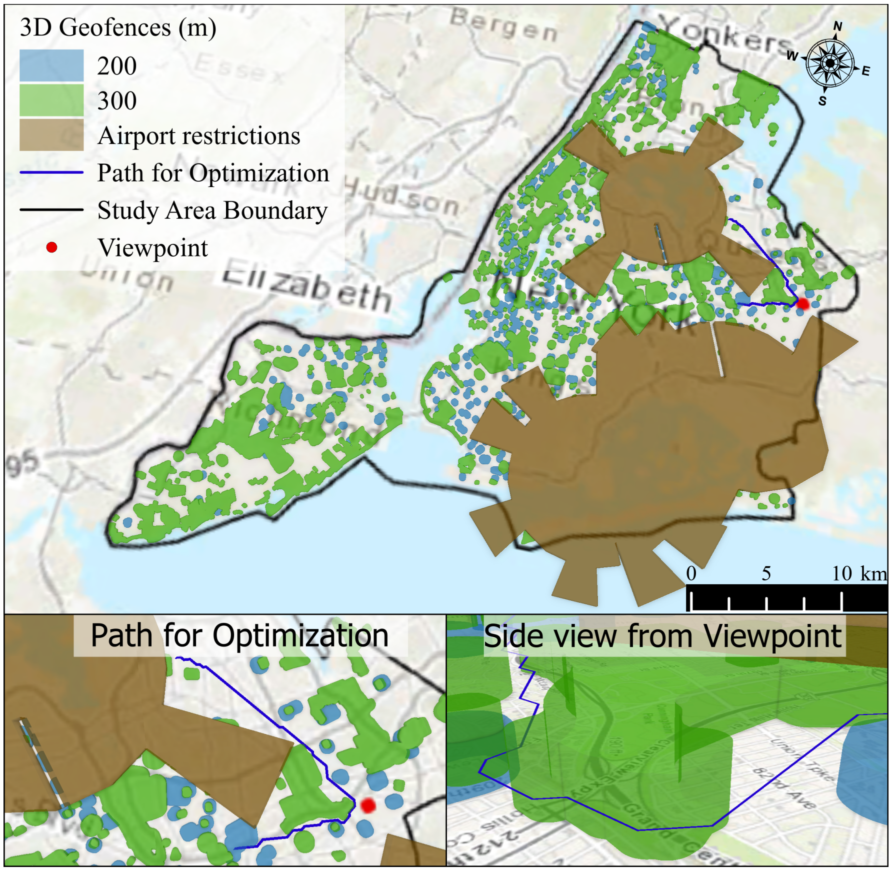

Fig. 3: Maps of the study area Manhattan, New York

Street Map [19]. Flight obstacle Maps from the Federal Flight Agency [20] and rooftop heights from Open Data NY [21] are obtained. The in-detail description and reasoning for choosing the vertical and horizontal restrictions as depicted in Tab. 1, as well as special cases, can be obtained from a previous work on modelling the geofences with GIS[2] [6]. The combination of the data sources allow the computation of a least flight height plane for the 3D routes. The resulting format of this plane is a georeferenced and interpolated surface derived with the method of Inverse Distance Weighting interpolation [22]. For improved performance, this surface is converted in this study case into a raster format called GRID [23] with one cell having the size 30x30 meters.

For a better comparison to flight paths which were obtained by GIS computation methods without any optimization process,an extract of previously obtained results are shown in Fig. 3. Furthermore, the existing results are used for the flight route initialization. The study area with the restricted air space and the path to be optimized is illustrated in Fig. 3. The blue path is one part of the least cost path calculation

[2] https://github.com/mohildemann/Urban-Air-Mobility-Routing

Table 1: Vertical and horizontal restrictions in meters

Landuse	Vertical Restriction	Horizontal Restriction
Airport	600	7620
Hospitals	300	300
Universities	200	300
Parks	300	100
Graveyards	300	100
Recreational Areas	300	100
Rooftops	152.4	-

from start to end points on the minimal flight height plane, which was calculated in the previous work [6]. The cost is the accumulated euclidian distance, the selected path is ≈19 kilometers long. The path is used to compare the quality of the obtained flight routes from the optimization. This part of the complete least cost network was chosen, because the shapes and positions of the restricted flight areas surrounding the path are complex on one hand, but leaves enough freedom for the optimization to find different solutions.

The path is also used for a seeding procedure. Seeding is an approach, where prior knowledge of the problems and its solutions exists and is used for generating the initial population for an evolutionary algorithm [24]. The existing path is already a valid solution, as the path does not interfere with the restricted airspace, connects a start with an end point and even minimizes the euclidian distances. This knowledge is used for the flight route optimization, as it is a known method for reducing the number of fitness evaluations by a faster convergence behaviour. The reduced number of fitness evaluations results in lower computational costs. Seeding is considered especially useful, if the initialization and evaluation of the solutions are expensive [24]. The proposed method of using geographical objects as representations for the optimization is an expensive computation. The trade-off is loosing variety by limiting the initial population [24], but this trade-off will be made to ensure convergence with a limited time frame and computation power. The proposed seeding procedure uses the existing least cost path as a centre of a 3D channel. The channel width is limited to 700 meters, 350 meters to both directions from the least cost path. The height of the channel depends on the distance from the minimal height to the maximal height at each position. Now, points are generated along the existing least cost path, at every 400 meters along the line. These points are then randomly moved to a location within 350 meters on a horizontal plane. Then, the minimum and maximum altitude of the new position are examined and the altitude is randomly assigned to a value within this range. If the point is moved to a location within the restricted flight area, the point is deleted.

The next step after generating valid solutions is the evaluation of each solution for the defined objectives flight time, energy consumption and noise. To compute these thee objective values, several data sources are required. The following listed input data in the same coordinate system[3] is needed for the validation and the evaluation of flight routes.

[3] Input data available at https://github.com/mohildemann/3D-Flight-Route-Optimization/ in the compressed geodatabase OptimizationInputs.gdb.zip

1. Transportation noise in raster format
2. Minimal height raster
3. 3D geofences in shapefile format
4. 3D line/2D line of an existing flight route

The minimal flight height plane and the restricted areas are sufficient information for computing the flight time and the energy consumption of a flight path. For being also able to compute the additional noise, the existing traffic noise is needed additionally. The data for a weighted, average sound level for the day from aviation and interstate road noise in the year 2014 is available at the United States Department of Transportation [5].

Furthermore, the flight characteristics of the aircraft flying the computed flight paths have an impact on the objectives. Many eVTOL configurations exist with different thrust types, among of them "Thrust, Lift and Cruise, Wingless, Hover Bikes and Personal Flying Devices, Electric Helicopters" [25]. In this study case, the wingless multicoptor eVTOL Ehang 184 [26] is compared to the vectored thrust eVTOL Lilium Jet [16]. The choice falls on the two categories, as the short-distance and long-distance performance vary the most in a comparison study between three eVTOL types [27]. The wingless multicoptor is more energy efficient in short distance missions whereas the Lilium Jet is more energy efficient and faster in long-distance missions. Furthermore, the chosen aircraft are the Lilium Jet and the Ehang 184 as the necessary aircraft parameters for calculating the energy consumption and speed were elaborated in that study by Bacchini and Cestino [27].

For this study case, the Python extension libraries of the licensed GIS desktop application ArcGIS Pro 2.4.1[4] are used for the data storage and data processing of the spatial data. The optimization program is developed in Python 3.68[5]. The Python environment from ArcGIS Pro 2.4.1 was used, which has all the required libraries for the optimization pre-installed. The whole code, the software dependencies and the required input data are available on GitHub[6]. The code for the optimization can be found in the directory "baselinerouting".

2.3 Flight path optimization with EA and GIS

2.3.1 Representation and initialization of 3D routes

The representation of the 3D flight routes is a sequence of georeferenced 3D points (Fig. 2), that result from the seeding procedure. The 3D points are randomly initialized within a 3D-channel with the seeding procedure, but the starting and ending points have a fixed position. Each point consists of the x and y coordinates and z as the height in meters above ground. The x, y and z values are randomly created within the 3D seeding channel. This is done by generating points along the existing shortest flight path. Every 400 meters along the shortest flight path, a point is ran-

[4] https://www.esri.com/de-de/arcgis/products/arcgis-pro/resources

[5] https://www.python.org/downloads/release/python-368/

[6] https://github.com/mohildemann/3D-Flight-Route-Optimization

domly moved within a maximal 350 meters range. Then, the z value is generated at random between the minimum and the maximum possible flight height of the current position.

In contrary to Genetic Algorithms with fixed-length bit strings, each genome is a real coded array with the x, and z-coordinates [10]. By definition, algorithms using fixed length real encoded representations are called Evolutionary Strategies, whereas Lee and Antonsson called real encoded solutions with variable lengths exG[28]. In this work, a variable length of genomes in the solutions is considered more useful than fixed lengths. In the optimization procedure we face the possibility, that the optimal flight routes are longer for one objective than for others. For example, in our scenario, flight routes above the street network with its corners are probable to are better for the objective of the minimal noise addition. On the other hand, following the street network is most often not the shortest way. Now, if we limited the solutions to a fixed amount of 3D points, we might also limit the ability to find the optimal solutions: A flight route might be optimal, considering only the objective of noise, where the 3D points are positioned above each intersection of the underlying street network. If the fixed number of points is lower than the number of intersections, the optimal solution is not attainable. On the other hand, a lower number of points might favor the other objectives, for example minimizing the flight time or energy consumption. The optimal solution for this objective might not require many points. It might even be better to have a lower amount of 3D points to decrease the amount of direction changes. By using a dynamic length of genomes, we can not favour any objective by the length of its genomes. Furthermore, we do not discard the possibility to find an optimal solution for each objective in the same run, as the amount of points can differ in the end.

Another reason concerns the adaptability of the solutions to the complexity of finding the optimal flight route. Depending on the density, placements and shapes of the restricted flight areas, different amounts of 3D points might be necessary to find the optimal route. If many small restricted areas exist in one neighbourhood of the city, more points might be needed to find a valid flight route through the more complex structure of the non-restricted airspace than in a neighbourhood with only one big restricted area. The strategy of a dynamic length therefore does not limit the solutions to perform only well in a specific level of complexity. Hence, it increases the generality of the method compared to a fixed-length approach.

Furthermore, each route solution also consists of a smoothed line which connects the 3D points (Fig. 2). The line is needed to validate a route, because straight connections of the points can intersect with the restricted air space, even if the points are positioned outside of the restricted air space. A polynomial approximation [29] of the line is necessary to make the solution more realistic, because the moment of inertia does not allow the eVTOL to fly around corners without stopping.

The array representation, the 3D points and the 3D line are intertwined (Fig. 2). If one of these three representations of one individual changes, the other two representations need to be updated accordingly. It is important to notice, that the only object manipulated with the genetic operators is the array representation. The computation of the mutation and crossover is done with the values within the array. If any element is manipulated, the points and lines are updated afterwards. The 3D points and lines are only manipulated during the validation process in the replacement process. If any point or line of a solution is replaced, the changes are transferred to the other two representations, too.

2.3.2 Validation of 3D-routes

The 3D-routes need to be constructed within the non-restricted airspace to be valid, as explained in Sec. 2.1. Therefore, the constraints for finding a feasible solution are respected by creating 3D routes, whose nodes (3D points) and connecting edges (3D line segments) do not intersect with the restricted regions. For being a valid solution, the following conditions need to be fulfilled:

- No 3D line segment intersects with the minimal flight height plane.
- No 3D line segment intersects with the 3D geofences.

The usual approach in evolutionary algorithms is the neglection of invalid solutions by assigning them the worst possible fitness value [15]. Nevertheless, in some scenarios it is appropriate to repair solutions instead of neglecting them [30]. In the proposed approach, the computation effort for producing a solution is high and the probability of violating a constraint is high. The covered area of 3D geofences is ≈ 41%, and the flight route is ≈19 kilometers long. In the probable event of a constraint violation at any point of the optimization, a good solution intersecting minimally with the minimal flight plane or a geofence can be lost. To counteract this, repair processes are proposed. If the solution is still invalid after a repair trial, the solution is removed.

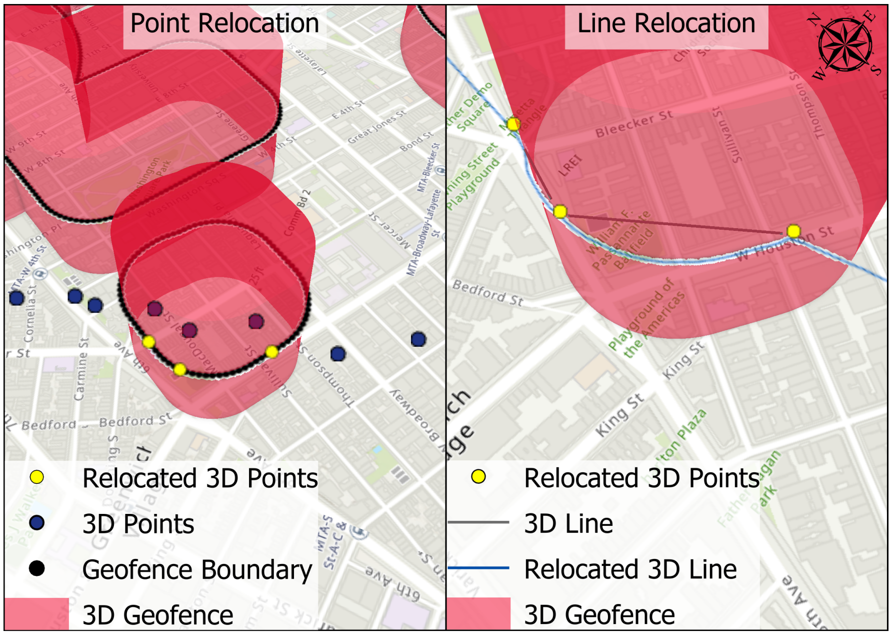

Fig. 4: Illustration of repairing points and lines intersecting with geofences

Two geoprocessing workflows are used for the repairments of the points and lines. The relocation of points in the restricted air space is illustrated in Fig. 4 (left), the relocation of lines in Fig. 4 (right). As 3D points are randomly initialized with a z value within the range of the minimum and maximum flight altitude, this condition needs to be checked when the x- or y-position of the point changes. If the z value is lower than the minimal flight height, the z value will be updated to the minimal flight height at that position. The second repair process needs to be executed if a 3D point lies within a geofence. For this case, additional reference points are created before the optimization process begins. The reference points were created in 10 meter distance from the border lines of the geofences. All 3D points, that are positioned in the restricted air space, are relocated to the x- and y-coordinate of the nearest reference point. The altitude of the point does not change. This means, that the altitude for the relocated points needs to be revalidated to assure, that the altitude for the new positions are higher than the minimum flight height and lower than the maximum flight height. The second repair function repairs intersections of 3D lines with the restricted airspace. Even if the 3D points are not positioned in the 3D geofences after the point relocation, the connecting line segments might do intersect with the restricted air space, as can be seen in in Fig. 4 (right).

Algorithm 1 Line repair

```
$[RoutePoints] = [Points]$                                   ▷ Points of flight route
$ip1, ip2 \leftarrow Intersect(FlightRoute, Geofences)$      ▷ Intersection points
$bp1, bp2 \leftarrow Near(ip1, PointBoundary)$               ▷ Two nearest geofence boundary points
$[way1], [way2] \leftarrow [Points]$                         ▷ Initialize empty arrays
while $ip2 \notin [way1, way2]$ do                           ▷ Stops if the second intersection point is reached
    $np1 \leftarrow Near(bp1)$
    if $np1 \notin [way1, way2]$ then
        $b1 \leftarrow np1$
        $[way1].add(bp1)$
    end if
    $np2 \leftarrow Near(bp2)$
    if $np2 \notin [way1, way2]$ then
        $b2 \leftarrow np2$
        $[way2].add(bp2)$
    end if
    if $[way1].last = ip2$ then
        $[Route].add([way1])$
    end if
    if $[way2].last = ip2$ then
        $[RoutePoints].add([way2])$
    end if
end while
$[RoutePoints] \leftarrow Reorder([RoutePoints])$
$Line \leftarrow PointsToLine([RoutePoints])$
return $[RoutePoints], Line$                                 ▷ Returns flight route with new elements
```

At first, the intersection points of the 3D line with the 3D shape are computed. From these intersection points, the nearest two neighbouring points from the geofence point boundary are added to two arrays. For each point, the nearest neighbour will be added to the array. Now, for every last point of each array, the nearest neighbour point that does not yet exist in any of the two arrays is added. This procedure is repeated until the second intersection point is added to one of the two arrays. The points in the the arrays are inserted into the point representation. In this way, the geofence is surrounded in both directions and the shorter way is used to be inserted to the point representation. If the 3D point representation is now converted to a line again by connecting the points, the line is not intersecting with the geofence any more. The repaired line is illustrated in Fig. 4 (right, Relocated 3D Line). In order to achieve this point to line conversion without crossings, the points need to be in the correct order. This is achieved by reordering the all 3D points after all point and line repairs, from the starting point to the end node. After this, the old 3D line representation is replaced by the repaired 3D lines. All changes are propagated to all three representations of the solution.

2.3.3 Evaluation of the 3D-routes

If the 3D points and 3D lines are repaired and all validation requirements are fulfilled, the next step is the evaluation of the individuals as can be seen in the flowchart in Fig. 2. For each of the three objectives, an appropriate fitness function is needed in order to quantify the quality of an individual for each criterion.

The objective functions depend on some variables, as the flight time, the energy consumption and the added noise do not only depend on the chosen flight route. They also change with the type and the flight characteristics of the aircraft, which will follow the flight path. The majority of the flight parameters of the two aircraft Ehang 184 and the Lilium Jet stem from the eVTOL configuration comparison study conducted by [27]. Some aircraft characteristics, for example the sound pressure, needed to be estimated, as these characteristics are not made available yet by the aircraft producers.

The general approach of computing the three objective functions is laid out shortly before going into the mathematical details. For computing the flight time of an aircraft, the maximum possible or allowed velocity and the length of the segment are needed to compute the velocity at each flight route segment. Corners along the flight route, maximum gravitational forces in corners for not feeling uncomfortable, flight speed limitations by the aircraft and lastly flight speed limitations by law are possible factors that limit the maximum velocity at each position. Furthermore, it needs to be calculated, whether maximum possible velocity can be reached at all with the acceleration speed of the aircraft. In order to calculate the energy consumption, a differentiation needs to be made by the type of aircraft flying the route. The Lilium Jet for example has fixed wings that produce lift at higher flight speeds, whereas the multicoptor from Ehang does not have wings. The computation therefore differs when the cruise speed is reached, which means the aircraft is neither accelerating, nor hovering nor decelerating. Apart from that, the variables are the same for both aircraft types. The aircraft and flight route characteristics together influence the required energy consumption. The last objective function of the additional noise also requires information from the noise level at ground, because flight routes are favoured where the noise level is high. The higher the existing background ground noise and the further the distance to the ground, the lower is the additional noise coming from the aircraft.

2.3.3.1 Preparation

Before the objective functions can be computed, some metrics need to be calculated in order to compute the flight time and the energy consumption. One of these metrics is the maximum flight speed at any position of the flight path. The maximum flight speed can be limited by three different factors: The maximum allowed flight speed by the flight authorities, the maximum possible flight speed by the aircraft and the flight comfort. The maximum possible cruise speed for the Lilium Jet is 252 km/h, and for the Ehang 184 it is 100 km/h [27]. We assume, that the flight agencies and city authorities will not allow air taxis that fly at speeds of 252 km/h in the urban areas, so in the computation we limit the maximum velocity to 100 km/h. The maximum flight speed to assure the flight comfort is more complicated. We assume, that similar flight acceleration forces as in traditional flight transportation are aimed for. To guarantee the flight comfort, a gravitational force equivalent (g_{Force}) of 1.25 is aimed for, which is a standard value for turns in passenger flights [31]. If the aircraft need to fly turns, the passengers are exposed to centripetal forces. As the flight route is no single circle, the centripetal force changes at each point of the route. In order to compute the centripetal acceleration at each position, an imaginary circle is computed. This imaginary circle is constructed at every point with its two neighbour points. The radius of the imaginary circle is used to compute the centripetal acceleration a_c and the resulting maximal velocity v_{max} for the maximum g_{Force} of 1.25 [31]. The formulas for the computation are expressed in Eq. 1, 2 and 3.

$$a_c = \frac{v^2}{r} \tag{1}$$

$$g_{Force} = \frac{a_c}{g_m} \tag{2}$$

$$v_{max} = \sqrt{g_{ForceMax} \cdot r \cdot g_m} \tag{3}$$

where

a_c	: Centripetal acceleration in $\frac{m}{s}^2$
v	: Velocity in $\frac{m}{s}$
r	: Circle radius in m
g_{Force}	: Gravitational force equivalent
$g_{ForceMax}$	: Maximum allowed g-force
g_m	: Gravitational acceleration of $9.81\frac{m}{s}^2$
v_{max}	: Maximum allowed speed at given radius and maximum g-force

The output is the maximal flight speed at each position, which is the minimal value of the maximal allowed legal flight speed, the maximum speed of the aircraft and the maximum speed to have a lower g_{Force} than 1.25.

2.3.3.2 Required flight time

Flying at the maximum flight speed at each position is not possible as the aircraft need to accelerate to this speed beforehand. An example: Even if the maximum possible flight velocity was 200 kilometers per hour 50 meters after leaving from the starting point position, it is not likely to accelerate the aircraft to that velocity within the acceleration distance. It would only be possible if the aircraft was able to accelerate with 27.7 meters per square second, which is neither realistic for air taxis nor comfortable for commuters. For deceleration the same physical laws apply. If the current speed was 200 kilometers per hour and the maximum speed in the next corner in 50 meters distance was 0 kilometers per hour, due to a 90 degree turning angle, the aircraft would not be able to decelerate in that short distance to 0. Therefore, the acceleration and deceleration speeds are necessary aircraft parameters and need to be considered.

The total flight time with the specified flight constraints is the added up flight time of all line segment:

$$t_{total} = \sum_{i=1}^{n} \frac{d_i}{v_i} \tag{4}$$

where

t_{total} : Total flight time
n : Number of line segments
d_i : Euclidian distance from current point to next point
v_i : Flight velocity at current segment

To calculate the flight time of each segment, we use the distances between the points and actual flight speeds. The aircraft parameters by [27] are needed for calculating the actual speed at each position. If the aircraft is slower than the allowed flight speed at the next position, the aircraft accelerates with an acceleration of $2\frac{m}{s}^2$. If the allowed flight speed of the next position is lower than the current flight speed, the aircraft decelerates with -$2\frac{m}{s}^2$.

Furthermore, it needs to be assured, that the distance to the next point is sufficiently big in order to decelerate to the allowed flight speed. If the distance is not big enough, the flight speeds of the previous points are readjusted. For example, if the required distance to decelerate with -$2\frac{m}{s}^2$ from the current velocity to the maximum allowed speed is 100 meter, all previous point velocities within the 100 meter distance are updated accordingly.

In Fig. 5 it is illustrated,how the positions of the vertices do not only affect the length of the flight route, but also have an impact on the flight time. Tight corners force the aircraft to decelerate to very low flight speeds in order to comply with the the maximal g_{Force} of 1.25. Flight route structures having these sharp corners will inevitably have a worse fitness for the objective flight time compared to routes without sharp turns.

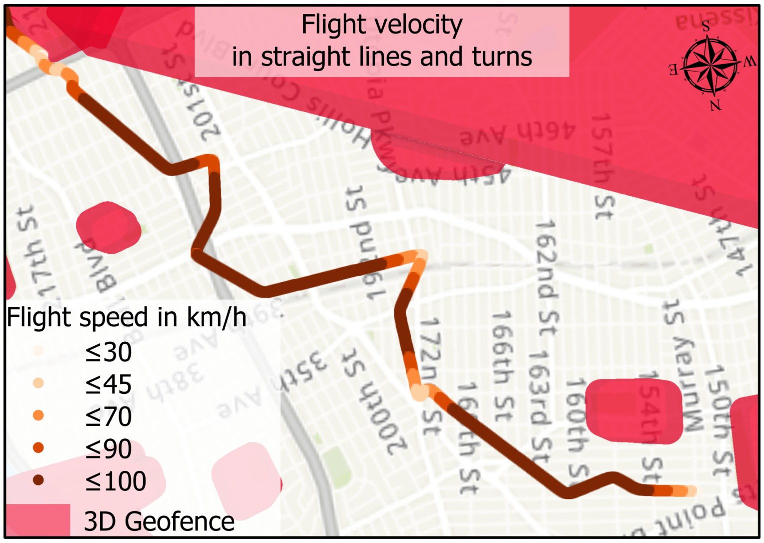

Fig. 5: Illustration of acceleration in straight segments and deceleration in turns

2.3.3.3 Required energy consumption

For calculating the required energy consumption for the flight, the aircraft type needs to be taken into consideration. The vectored thrust aircraft can generate more lift with increasing flight velocity with their wings. On the other hand, the wingless multicoptor generate less additional lift. This leads to a differentiation for the computing the energy consumption [32]. In our simulation, the power required for acceleration, hovering and deceleration does not change for both aircraft types in order to be able to use the results from Bachhini et al. [27]. The difference lies within the calculation of the energy at cruise speeds, as aircraft with wings have a better lift/drag ratio with increasing flight speeds. The flight speed for an optimal lift/drag ratio is 252 km/h for the Lilium Jet, which results in a required energy of 28 kW [27]. For the flight speed of 100 km/h, the required energy is 34.6 kW. The changing energy consumption depending on the flight speed explains the x in Tab. 2, which is a place holder for the unknown required energy for different velocities. The multicoptor on the other hand has no wings and can only produce lift with its' rotors. Hence, the calculation differs by the aircraft type. The computed aircraft performances of the E-Hang 184 and the Lilium Jet are:

As indicated in Tab. 2, the required energy for landing, hovering, acceleration and deceleration is constant. The more complicated case is the calculation in cruise mode, where the generated lift and a reduction in the induced drag reduce the power consumption [27]. The drag polar for the Lilium Jet in Tab. 2 is crucial for the calculation of the actual required power in cruise mode. Additional required parameters are the air density and the gravity. These are set to the standard values of $1.225\frac{kg}{m^3}$ and $9.81\frac{m}{s}^3$. Now, with the variables flight velocity and the flight angle, the required power can be calculated for each flight route segment by following the

Table 2: Aircraft parameters for calculation of energy consumption from Bachhini et al. [27]

	E-Hang 184	Lilium
eVTOL type	Wingless Multicoptor	Vectored Thrust
Cruise power in kW	34.6	28 / 36 / x
Hover power in kW	42.7	187
Acceleration energy in kW	42.7	187
Deceleration energy in kW	42.7	187
Acceleration speed in $\frac{m}{s}^2$	2	2
Deceleration speed in $\frac{m}{s}^2$	-2	-2
Wing area in m^2	0	3.6
Empty weight in kg	168	490
Drag polar	-	$0.0163 + 0.058C_L^2$

calculus of Eq. 5-11 [32]. The velocity is derived from the previous step described in Sec. 2.3.3.2, and the ascending/descending angle is computed for each point to the next point. Fig. 5 illustrates, that all flight route segments which are not dark red, require the aircraft to accelerate or decelerate with the corresponding acceleration or deceleration energy. The required energy in the dark red segments differs with the ascending/descending angle for each segment. By summing up the required power of all segments, the desired objective function for getting the total energy consumption in kWh is complete (Eq. 12).

In the chosen set-up, the Lilium Jet faces disadvantages compared to the E-Hang 184, as the type of aircraft is designed to operate at higher speeds than 100 km/h. The maximum velocity of 100 km/h was selected, as in the early phases of air taxi operations it is likely that low flight speeds are established. The maximum legal speeds for air taxi operations in urban areas have not been defined or at least have not been published yet by flight authorities. Nevertheless, the optimization can be run with different parameters of the assumed maximum legal flight velocity.

$$L_C = W * cos(\gamma) \tag{5}$$

$$L_{CL} = \frac{L_C}{\frac{1}{2}p^2 S} \tag{6}$$

$$C_{DC} = C_{D0} + K * C_L^2 \tag{7}$$

$$D_C = \frac{1}{2}pV^2 SC_{DC} \tag{8}$$

$$T_{RC} = W\sin(\gamma) + D_C \tag{9}$$

$$T_{RC} = W\sin(\gamma) + D_C \tag{10}$$

$$P_{RC} = \frac{T_{RC}}{1000} \tag{11}$$

$$P_{RC_{total}} = \sum_{i=1}^{n} P_{RC_i} * \frac{d_i}{v_i} \tag{12}$$

where

L_C	: Required lift
W	: Weight of airplane in N
γ	: Angle of climb/descent in °
L_{CL}	: Lift coefficient in climb
S	: Wing area of aircraft in m^2
V	: Flight speed in $\frac{m}{s}$
p	: Min. Air density $\frac{kg}{m^3}$
C_{DC}	: Drag coefficient in climb
K_C	: Drag factor
C_{D0}	: Total zero-lift drag
D_C	: Drag in climb in N
T_{RC}	: Thrust required in N
P_{RC}	: Power required in climb in kW
$P_{RC_{total}}$	: Total required power of flight route
n	: Number of line segments
d_i	: Euclidian distance from current line segment i
v_i	: Flight velocity at current segment i

2.3.3.4 Added Noise

The last objective function computes the added noise to the existing background noise at ground. The Environmental Protection Agency [33] and the World Health Organization [34] support the approach of planning flight routes above areas where the noise level is already high. This leads to less added noise in quiet areas. For computing the last objective function for the criterion least added noise, three steps need to be processed, which are adopted from and explained in full detail by Kinsler et al. [35]. The proposed approach only calculates the added noise directly underneath the aircraft, despite the fact that sound waves move in space as expanding spheres and therefore affect underlying areas and not only the underlying points. We only compute the added noise perpendicular from the aircraft to the surface rather than considering all the areas that the sound waves might reach. In this work, the behaviour of the noise is simplified in order to keep the objective function simpler and computationally less expensive. Furthermore, for a better and faster understanding of the following equations, the sound pressure level is explained shortly. Due to the limited range of perception of the human ear to experience sound pressures, the unit for sound pressure decibels (dB) was designed to cover the noticeable range [35]. The logarithmic based unit does not allow simple comparisons, e.g. adding several noise sources, without logarithmic rescaling.

The first step is to calculate the sound pressure arriving at the ground. For this calculation, the noise emission of the aircraft is required. These values are not made public from the aircraft manufacturers, therefore the targeted noise level of Uber Elevate is used as replacement. Uber Elevate is the only available source to mention the noise level targets [36]. Their goal is a maximum of "62 dB at 500 ft altitude". This translates to a noise pressure of 100 dB in a reference distance of one meter. By filling the variable r in Eq. 13 with the current flight height we get the perceived noise level at ground from the aircraft.

The next step is to combine the noise emission by the aircraft with the background noise at the ground. The used data for the background noise is the average daily sound level from aviation and interstate road noise. The data is from 2014 and made available by the United States Department of Transportation [5]. By combining the aircraft and background noise by inserting corresponding the noise levels at variable $L_{P.i}$ in Eq. 14, the total noise level $L_{P.Tot}$ is computed. At this point, it is possible to determine how much noise the aircraft added at ground. It is achieved by subtracting the background noise from the total noise in Eq. 15. In this equation can be observed, that the noise addition is higher if the background noise is lower. Therefore, the proposed objective function complies with the idea of preferring areas with a high background noise.

$$L_P = L_m - 20\log_{10}\frac{r}{r_m} \tag{13}$$

$$L_{P.Tot} = 10\log\left(\sum_i 10^{0.1L_{P.i}}\right) \tag{14}$$

$$L_{P.Source} = 10\log\left(10^{0.1L_{P.Total}} - 10^{0.1L_{P.Background}}\right) \tag{15}$$

where

L_P : Sound pressure at the ground[7]
r : Distance to ground
r_m : Reference distance (1 m from aircraft)
L_m : Sound level at reference distance (100 dB)
$L_{P.i}$: Each noise from source i measured at ground
$L_{P.Tot}$: Combined noise pressure from background and aircraft noise
$L_{P.Source}$: Added noise pressure from aircraft at ground

The average added noise level is calculated by averaging all added noise levels at each point at the flight route.

2.4 Multiple objective evolutionary algorithms for 3D-routing

In multi-objective optimization, multiple objectives are optimized simultaneously. In the trade-off situation, solutions that can not be improved for one objective without discarding the other objectives are called non-dominated solutions. Non-dominated solutions are all optimal compromises between the objectives [10]. A solution is

[7] all sound pressure units are expressed in $decibel(dB)$

called non-dominated, if there is no other solution which is better in at least one objective and which is in the same time not worse in the other objectives [37]. A solution is dominated by another solution, if the other solution is better in at least one objective and equally good in all other objectives [37]. These solutions form a Pareto front. The non-dominated flight routes are derived in every iteration of the evolutionary algorithm.

2.4.1 Non-dominated sorting algorithm II

For solving the multiple objective optimization problem, the Non-dominated sorting algorithm II (NSGA II) is chosen. By using this strategy, multiple pareto-optimal solutions can be found within the population in one simulation run. This is not the case for all optimization techniques, and is considered as an advantage of NSGA [10]. The flowchart of the NSGA II is illustrated in Fig. 2 in the second loop. The details, how the algorithm works in every detail is outlaid by its developers Deb et al. [14]. The NSGA II is a widely used procedure and therefore not repeated in every detail. A description of the process can be found in the Appendix. The procedure of the NSGA II, in short, takes in all solutions from the parent and child generation and computes the non-dominated solutions which form the Pareto front.

2.4.2 Evolutionary operators

The evolutionary operators determine, how the individuals are selected and manipulated to evolve to better solutions. The selection process is adopted by the Deb et al. [14]. The crossover operation is a standard approach, while the mutation mechanics bases on the idea of Peng et al. [13], who applied Genetic Algorithms for 3D routing.

2.4.2.1 Selection

The selection of the individuals for the crossover and mutation operations is done with the so-called tournament selection [10]. From the population P, a subgroup is selected, the $TournamentGroup$. The selection to get into this group is random. From the $TournamentGroup$, the individual with the highest fitness wins. All individuals, even the tournament winner, can be reselected in the following tournaments. In this case of a multi-objective optimization, the selection bases firstly on the non-domination rank. In case, more than one solution belongs no the best rank, the individual with the highest crowding distance wins the tournament. The tournament winner will proceed to the crossover and mutation. The selection process is repeated until the population size is reached. The tournament size determines the selection pressure: the higher the number of individuals in the tournament, the smaller it is for low-ranked individuals to win the tournament [10].

2.4.2.2 Crossover

The crossover mimics the biological reproduction. The idea behind this process is to combine the genes of the parent generation with the potential of getting a better individual. This bases on the fact, that both parents have had special properties that improved their probability of being selected for reproduction [10]. The chosen crossover technique is an adapted version of the n-point crossover [38]. For the crossover, the genes of two selected tournament winner from the selection process are combined, who are referred to as parents. The produced offsprings inherit the combined elements of the parents. At *n* randomly chosen positions of the first parent, genes are split. In cases, where the presentations of individuals have a fixed length, these randomly chosen positions of the second parent correspond to the position of the first parent. In our approach, the number of points can differ with each solution. Therefore, the recombination process is slightly adapted.

1. The *n* randomly chosen positions are chosen in a range between 1 and the length of the shorter parent -1. The last position and first position are not exchanged as they are always the starting and ending point of the flight route.
2. For each recombination point *n* of the first parent, the nearest flight route point of the second parent is sought for with a GIS Near Point Analysis from ArcGIS Pro. These points are the beginning points of the point exchanges.
3. With both recombination positions of the parents, the offsprings are produced by exchanging the parent segments. Up to the first recombination point, children 1 gets the segments of parent 1, and children 2 gets the segments of parent 1. After the first recombination point children 1 gets the segments of parent 2 and children 2 gets the segments of parent 2. This changes at each recombination point *n*.
4. The points of the children are reordered. All changes are propagated to the array, point and line representations.

2.4.2.3 Mutation

Mutation grants the possibility to generate diversity by manipulating the genes passed on by the parent generation. This yields the potential of introducing characteristics, which do not exist in the population but can be helpful to survive in the environment [10]. The used mutation technique for the optimization of 3D flight routes bases on the idea of Peng et al. [13], who also used Genetic Algorithms for 3D routing. They split the mutation process into deletion, insertion and the disturbance of genes, in our case points. These processes were slightly adapted to increase the probability of high quality solutions.

The selection of inserting and deleting point is therefore chosen to be not probabilistic, but deterministic. The criterion is the point distance to the neighbours for selecting the points for deletion and selecting points where a point is inserted in between. The reason for this approach is the high area of restricted airspace. Higher distances between the points increase the probability of a line intersecting with the geofences. Therefore, the distance between the points shall be equalized stepwise with the insertion and deletion. This objective is achieved by deleting the points with the smallest average distance to their neighbour points. For the insertion process, the points are sought for with the highest distance to the neighbour point. The inserted points are be positioned between the neighbouring points.

The disturbance described by Peng et al. [13] is a point position change in a predefined radius. The position change of single points lead to the creation of sharp turns, which leads to a deceleration (Fig. 5). As this inevitably leads to a decreased fitness, we propose an adaptation of the point perturbation. Rather than manipulating a single point, a group of neighbouring points is also manipulated. To manipulate these points in a way that will not lead to the mentioned negative side effect of sharp turns, the points are positioned along a transformed sine curve. In Fig. 6, the sine mutation is illustrated and the pseudo-code for this can be found in Alg. 2.

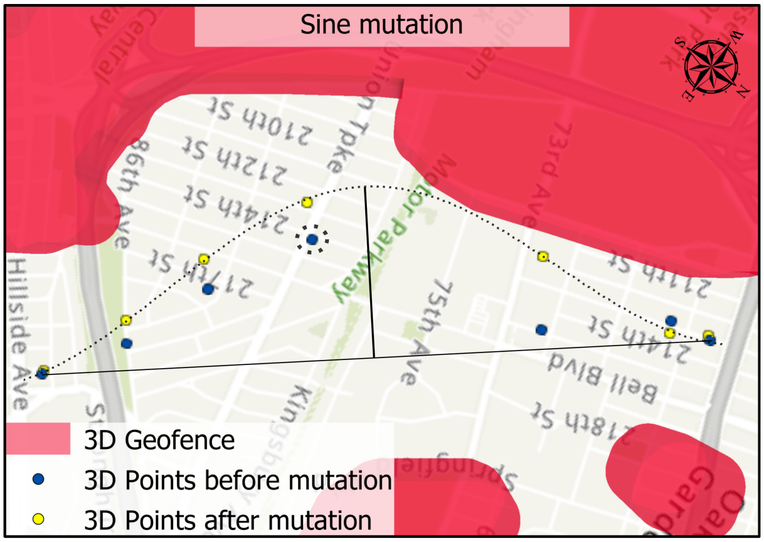

Fig. 6: Illustration sine mutation with a group size of 5

The sine mutation basically determines the neighbours within the specified group size and fits all of these to a sine function. The sine function is designed to build an arc between the starting and end node of the affected points. The maximum amplitude of this arc is determined by the parameter maximum disturbance distance. The actual amplitude is defined randomly for each mutation group and can vary from the maximum disturbance distance to its negative counterpart. Therefore, the possible mutation space for each mutation group is an ellipse from the starting point to the ending node. Having Fig. 6 as an example, the randomly chosen mutation point is marked. The random disturbance distance is illustrated as the perpendicular line of the connection line from the start to the endpoint. If the random disturbance distance was negative, the sine curve would be the reflected function. If the random disturbance distance was 0, the points would be relocated to the connection line from the start to the endpoint.

Algorithm 2 Sine Mutation

```
$input[Points]$                                          ▷ Flight route points of solution
$maxdist \leftarrow globalparameter$                     ▷ Parameter Max Disturbance Distance
$pm \leftarrow globalparameter$                          ▷ Parameter Mutation Probability
for Point in [Points] do
    if $Random(0, 1) < pm$ then
        $RandomDisturbance \leftarrow Random(-maxdist, maxdist)$
        $StartPoint \leftarrow Point - Floor(Groupsize/2)$
        $EndPoint \leftarrow Point + Ceil(Groupsize/2)$
        for i = StartPoint +1 : EndPoint -1 do
            $x \leftarrow Distance(i, StartPoint)/Distance(i, EndPoint)$
            $y \leftarrow RandomDisturbane * (0.5 * sin(2\pi * (x - 2\pi)) + 0.5)$
            $i.x = i.x + y$
            $i.y = i.y + y$
        end for
    end if
end for
return $[Points]$                                        ▷ Returns updated points
```

2.5 Visualization and Decision making with non-dominated solutions

The visualization and the decision-making of the non-dominated solutions in multi-objective optimization is a research area of its own [39]. This work focusses on the proposed algorithm, and not on the decision-making and visualization. Nevertheless, several components of possible visualization and decision-making techniques are needed for interpreting the convergence behaviour and for comparing different results with different optimization configurations.

The first object of interest is a visualization, that illustrates the optimization convergence towards each objective with each generation. This can be visualized in line plots, that depict the fitness values of the solutions. Two different metrics are usually of interest, the fitness value of the best solution and the average fitness value of the generation [40]. That line plot needs to be computed for every objective. In order to compare the different convergence behaviour towards all objectives, the fitness values are normalised and rescaled to a range from 0 to 1. In addition to the separated line plots for each line plot, a 3D scatter plot complements the visualization of the convergence behaviour. The fitness values for each objective are plotted on the three axis, as visualized in the upper right half of Fig. 7. This visualization technique is able to convey an impression about the shape of the Pareto front, because the positions of the non-dominated solutions depict the compromise of each solution [40]. Scatterplots can only be used for two or three objectives.

The information of the position in relation to all considered objectives helps to get an impression about the trade-off situation for each solution. This can be plotted with a technique called radial visualization [40]. The relative quality of the normalised fitness related to each objective is mapped to a position in a circle. Individuals with approximately equal fitness values for each objective will lie close to the center, whereas individual which do have one or two fitness values greater than the others lie closer to the objective position on the outside of the circle. In the case of an optimization with three objectives, three extreme positions form an equilateral

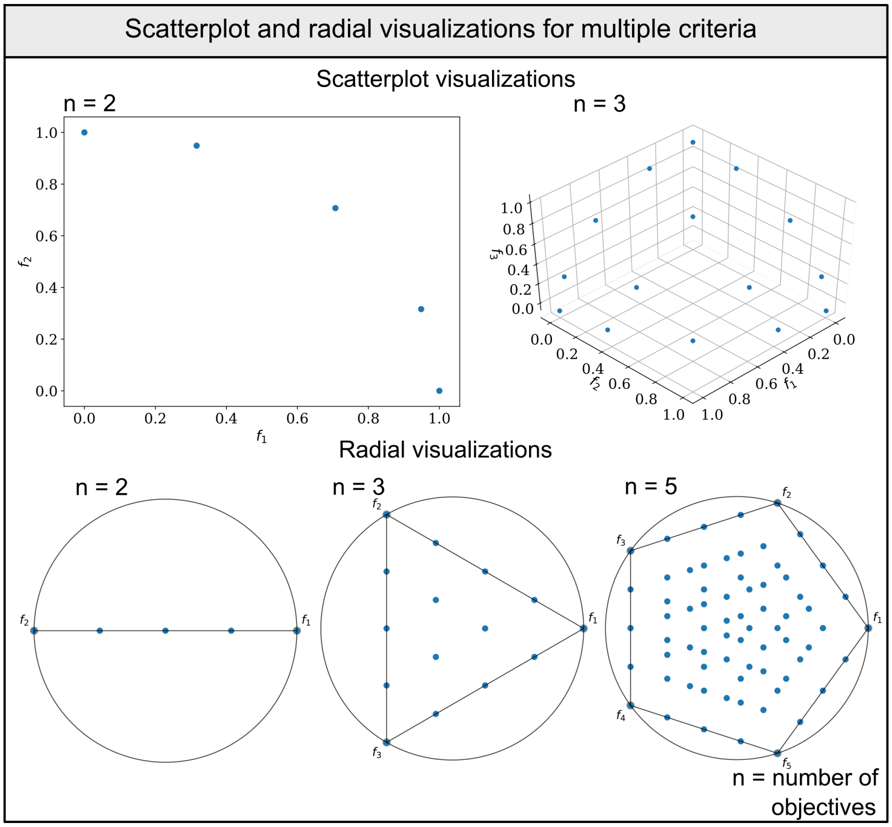

Fig. 7: Scatterplots and radial visualization for multiple criteria problems. Created with optimization problems and visualization tools from pymoo

triangle on the circle, as illustrated in the lower half of Fig. 7 with a number of three objectives. The extreme positions illustrate solutions that favour only one objective while neglecting the other objectives. All compromise solutions lie within the built triangle.

The last visualization is a map of the resulting 3D flight routes. The challenge in the map visualization is to maintain the readability of the map. In order to maintain the readability and reduce the amount of map elements, not all solutions are mapped. The selection of the flight routes to be mapped belongs to the decision making process. The difficulty mainly lies within choosing one solution based on three different criteria. Many decomposition methods exist for doing that selection, for example reference point direction methods or direction vectors. Some of the most often used methods are explained in detail in the works of Korhonen and Laakso [41], Deb and Kumar [39], or Ma et al. [42]. In this work, for reasons of simplicity, the extreme solutions and the compromise solution with equal importance on all criteria are selected with the weighted-sum method [43]. This methods sums up the normalised fitness values for each objective and multiplying every value with a weight between 0 and 1. In this study case of a minimization problem, the selected solutions is the solution with the minimum summed up weight. The extreme solutions are the optimal solutions for the singly objectives, so the solution with the shortest flight time, the lowest energy consumption and the lowest added noise. The weights for the selection for these values are therefore 1 for the favoured objective and 0 weight for the others. The compromise solutions in this case is calculated by summing up all normalised objective values with the same weight. The solution with the lowest summed up objective values is the selected compromise solution.

3 Results and Discussion

The produced results with the proposed methodology (Fig. 2) are discussed with different focuses. At first, the best optimization configurations are sought for getting the best results. With the best configuration, the optimization is run for both considered aircraft types, and the results are visualized with the proposed visualizations (Sec. 2.5).

3.1 Optimization configurations

The possible parameters for the proposed optimization of the example flight route are versatile. The selection process of the parameters is often a trial and error process and is problem dependent [15]. For the proposed 3D flight route optimization, the point group disturbance in a sine curve is compared to the single point disturbance of the mutation developed by Peng et al. [13]. Subsequently, different optimization parameter configurations are compared for getting the best parameters for the final optimization runs.

3.1.1 Comparison of mutation types

The flight route mutation from Peng et al. [13] and the proposed adaptation of it are compared by their performances. The adapted version with a disturbance along a sine curve instead of single point disturbances is compared in four different pseudo-random optimization runs, using the flight characteristics of the Lilium Jet. The sine mutation group size was set to 4 and the maximum disturbance distance to 120 meters. The results of that runs are composed in Fig. 8. The average fitness of the best solutions for each objective per generation are shown, including the standard deviation as the area around the line. The graph shows, that the convergence to the lowest energy consumption is very similar with both types of mutations. The Pearson correlation coefficient between the means is 0.95, which underlines the observation. Also the developments of the shortest flight times correlate strongly with a coefficient of 0.93. The fitness is worse in the first 7 iterations of the optimization, but is better in average by 0.3 seconds after 20 iterations. The most different development is observed for the lowest added noise. Even if the results of the last iteration are less than 0.1 decibels, the sine curve mutations found the best solution already at iteration 10.

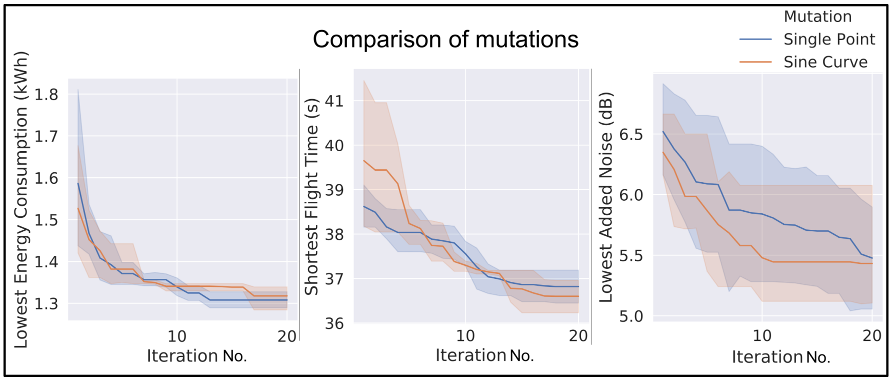

Fig. 8: Comparison of objective value improvements over GA iterations between sine mutation and single point mutation with 4 repetitions. The line represents the mean and the shaded area represents 95% confidence intervals

The graphs show, that the sine mutation produced better results for two of the three objectives, but only marginally better. In order to further understand the impact of a disturbance along a sine curve, the parameters sine mutation group size and the maximum disturbance distance are configured in the parameter search.

3.1.2 Parameter search for optimization

Due to the computation times, the hyper-parameter search process was kept small with seven repetitions. Also, the population size was kept constant with 12 individuals. The parameters for the seeding process are constant, too. Every 400 meters along the initial flight route line a point is randomly moved within a maximal 350 meters range. The flight characteristics of the Lilium Jet were used for the parameter search (Tab. 2).

The parameter ranges are composed in Tab. 3. Due to the high computational effort and the resulting low number of possible parameter configuration evaluations, the results of the parameter search can not be supplemented with a statistical significance test. We also can not analyse with certainty, how sensitive the optimization really is to the chosen parameters, because the true optimum is not known for any of the objectives.

Fig. 9 shows the distribution of the results for each chosen parameter set. Most results of the best solutions for the objective flight time range from 222 seconds to 227 seconds. Only one run with 242 seconds differed more than 10 seconds from the average. The range of the results concerning the objective energy consumption is higher, when it is compared by the range of the results to the reference point of zero. The results differ from 13.44 to 18.16. Lastly, the biggest differences ranges

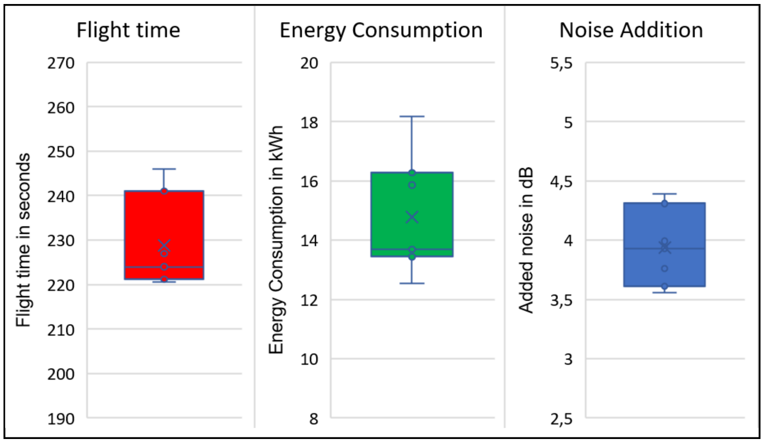

Fig. 9: Results of parameter search with 7 runs, a population size of 10 and 18 generations each. Point representing each run, cross representing the median result, whiskers representing 1.5 times the interquartile range

from 3.61 to 4.39. An important observation is the fact, that none of the results from the parameter search was an outlier.

The choice for the best parameter set was done by choosing a solution, which lies in the interquartile range for all criteria. The second criterion from the selection was to identify the solution with the smallest sum of all normalised fitness values. In regard to these two selection criteria, the best choice from the parameter sets produced a solution with a flight time of 224 seconds, 13.48 kWh energy consumption and 3.99 dB noise addition. The parameters of the selected parameter set-up are illustrated in Tab. 3 in the column "Selected parameter set-up".

Table 3: Parameter Search

Parameter	Minimum	Maximum	Different values	Selected parameter set-up
Selection pressure	25%	33%	2	33%
Number of crossover points	3	5	3	7
Crossover probability	50%	90%	4	90%
Mutation probability	25%	50%	5	50%
Sine mutation max disturbance distance	80m	120m	2	80m
Sine mutation group size	4	6	3	6
Sine mutation insertion and deletion	20%	40%	3	20%
Sine mutation disturbance	20%	60%	3	20%

The best results for of the parameter search were obtained with the maximum limits of the crossover probability (90%) and mutation probability (50%). The literature suggests high crossover probabilities [44], hence the result of the highest crossover probability is not surprising. On the other hand, literature also suggests small mutation probabilities, Schaffer et al. [45] even suggested mutation probabilities of less than 0.05%. They argued that higher probabilities add randomness, so that the optimization resembles a random search and therefore endangers a stable convergence behaviour. Nevertheless, the parameter choice is problem-dependent [15] and the used mutation method was adapted to find better solutions in the 3D flight routing problem. The proposed mutation method does not only add random elements, but even reduces the probability of being invalid by inserting points in parts of the flight routes where the distance between the points are high. Furthermore, the point disturbance step in the proposed sine mutation changes the position along a sine curve. The sine mutation rather supports the search for smoother flight routes than introducing random flight route points. These characteristics of the mutation are probably the key factors, why high mutation probabilities lead to better results. The possibility, that smooth mutations are a factor of generating good solutions is also expressed by the smaller maximum disturbance distance in the sine mutation with 80 instead of 120 meters. Also, the best sine mutation group size was the maximum possible group size of 6. A higher group size combined with a smaller maximum disturbance distance lead to sine curves that have a lower amplitude in relation to their length. It means, that the mutated flight route segments are less curvy. The last variable parameters were the sine mutation insertion, deletion and disturbance probability. The percentage expresses the number of points which are mutated from all flight route points. In the best parameter configuration, both are the minimum values of 20%. So all together, the mutation configuration that creates the best solutions are high mutation rates with small and smooth adaptations to the flight routes.

3.2 Convergence of the 3D flight route optimization

After having found a parameter set-up which produces satisfactory results, the optimization is repeated with a bigger population size of 30 for both eVTOL aircraft types. The higher population size improved the quality of the solutions in all criteria compared to the parameter search with a population size of 12 for the Lilium Aircraft. The shortest flight time is 211.45 seconds compared to the shortest flight time of the parameter search with 220.65 seconds. The same applies for the lowest energy consumption (10.8 kWh compared to 12.55 kWh) and the lowest added noise (3.51 dB compared to 3.56 dB). The trade-off to the improved results is a 6 times higher computation time. Fig. 10 shows the optimization convergence behaviour for both aircraft types. The best solutions for each objective and the average fitness of the non-dominated solutions of each generation are illustrated. In order to visually compare the convergence behaviour of the three different objectives, the fitness values were normalised to values between 0 and 1.

The shortest flight time for the Lilium Jet was already found after 4 generations, whereas the solution with the lowest energy consumption was found after 10 generations and the lowest added noise after 15 generation. In combination with the average fitness of all non-dominated solutions, the conclusion can be drawn that the optimization has converged the most to an optimum for the objective shortest flight time. The average fitness of all non-dominated solutions does not change any more

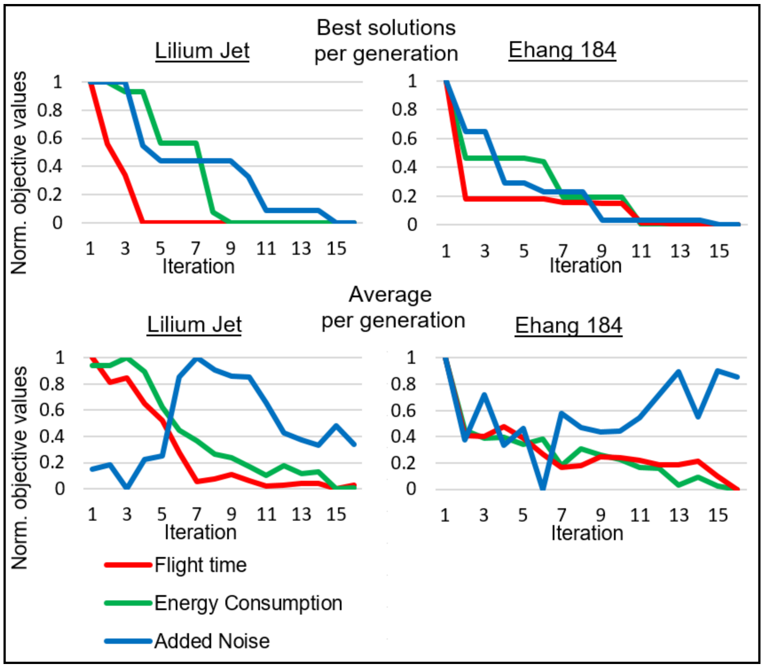

Fig. 10: Converging behaviour over the generations of best solutions and average objective values (normalised). Results for Lilium Jet (left) and Ehang 184 (right)

after iteration 7. On the other hand, the average quality development of the solutions for the objective lowest energy consumption decreases steadily until generation 11. The last objective's convergence behaviour of the lowest added noise stands in contradiction to the other observed developments. Until generation 7, where the average fitness of the shortest flight time already stagnates, the average fitness of the lowest noise decreases. After the seventh generation, the average fitness improves again. Even in the last generation, the average quality for the objective noise is worse than in the beginning. The curve trend indicates that the optimization has not converged yet to an optimum for this objective. The Pearson correlation coefficients r underline the observation: While the correlation coefficient for the average quality of the non-dominated solutions for the energy consumption and flight time is very high with r = 0.97, the correlations to the average added noise is anti-correlating to both the flight time (r = -0.66) and energy consumption (r = -0.54).

We can observe a similar behaviour for the second eVTOL type Ehang 184.Even if the developments of the best solutions for each objectives are more similar, the development of the average fitness regarding all non-dominated solutions show again a high correlation between flight time and energy consumption (0.95), the quality improvements for the best solutions in flight time and energy consumption even occur in the exact same generations 2, 6 and 10. On the other hand, the results of the objective lowest added noise do not seem to be related at all ($r_{flighttime,noiseaddition}$ = 0.1 and $r_{energyconsumption,noiseaddition}$ = -0.05).

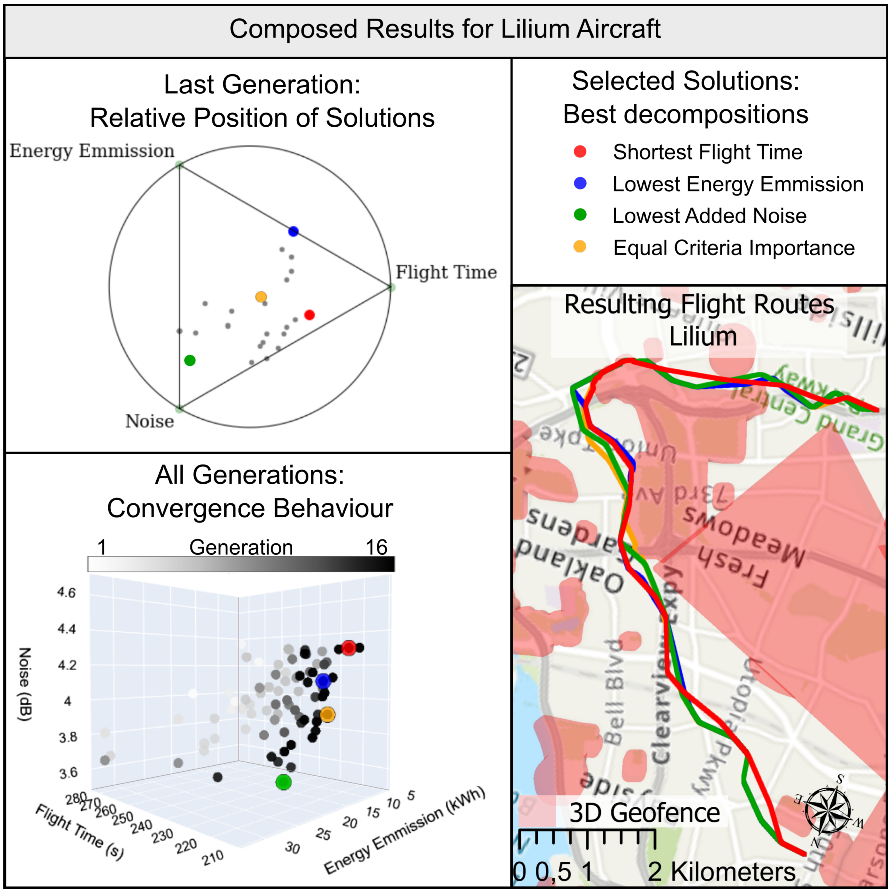

Fig. 11: Optimization results for the aircraft Lilium Jet

The composed results for the Lilium Jet in Fig. 11 and for the Ehang 184 in Fig. 12 illustrate the optimization process and results in three additional ways: the convergence behaviour towards the three objectives is illustrated in a 3D scatter plot over all generations. The relative positions of the non-dominated solutions of the last generation towards each objective are depicted in a Radial Visualization. Lastly, four selected solutions of the last generation are presented in a map. The best solution for each objective is marked, as well as the compromise solution with equal weights towards each objective. These four solutions are marked in all three visualizations for better comparability and interpretability. The results in the scatter plot for the Lilium Jet and the Ehang 184 show, that the quality of the non-dominated solutions over the generations improves steadily. Furthermore, the number of non-dominated solutions increases over time, which is the desired outcome of the optimization for a better decision making. The more non-dominated solutions were produced, the more compromise options the decision maker has for the subjective selection of a single solution [46].

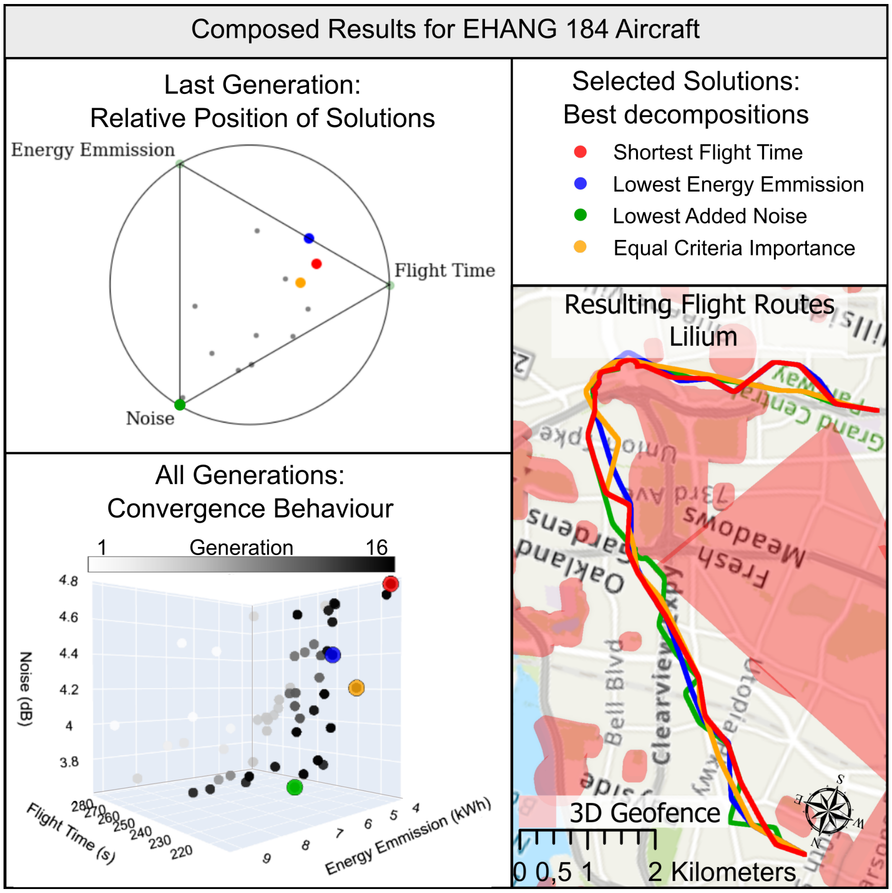

Fig. 12: Optimization results for the aircraft Ehang 184

The compromise solution in the optimization run for the Lilium Jet is located in almost equal distance to all the extreme points of the objectives, as well in the radial visualization as in the 3D scatterplot visualization. The radial visualization for the Ehang 184 reinforces the impression, that the convergence behaviour towards the objective of minimal noise addition differs more to the two other objectives than in the optimization with the Lilium Jets' flight characteristics. The solutions of the lowest energy and the shortest flight time are neighbour points in the radial visualization, which corresponds to the observed high correlation coefficients during the optimization process. The position of the lowest noise stands in high contradiction to the similarity of the shortest flight time and lowest energy consumption. The conclusion from the observation is, that the two objectives shortest flight time and lowest energy consumption are more related to each other than to the objective noise.

4 Discussion

This section discusses, whether the aim of optimizing geographically represented 3D flight routes with Evolutionary Algorithms was achieved. The previously described results are used as a basis for considerations that relate to 3D flight route optimization in a next step, including implications for the urban air space management. Finally, the implications for 3D route optimization in general are emphasized.

4.1 Discussion of the results derived from the study case

The first implication of the observed convergence behaviour in the 3D scatterplot in combination with the previously discussed Fig. 10 is, that the optimization of geographical objects towards multiple objectives and multiple constraints in 3D flight route optimization work with the proposed methods. The 3D scatterplots (Fig.11 and Fig.12, Illustrations “All Generations: Convergence Behaviour”) show, that for both aircraft the desired region in the lower-right corner with the shortest flight times, lowest added noises and lowest energy consumptions are populated by the non-dominated solutions over the generations. Compared to the least cost distance path, which we aimed to optimize, all objectives were improved by at least 36% for both aircraft types(Tab. 3). The biggest positive impact was the optimization of the energy consumption with a decrease in energy consumption of more than 70%. This underlines the positive impact of an optimization process of existing least cost paths derived with GIS algorithms. The finding is a motivation for further work of optimizing 3D routes in a GIS environment.

Table 4: Optimization results compared to Least Cost Path

	Flight Time	Energy Consumption	Added Noise
Least Cost Path (Lilium, Ehang)	362 s, 362 s	52 kWh, 13 kWh	5.61 db, 5.61 db
Lilium Jet: Optimal solutions	211 s (-42%)	5.39 kWh (-90%)	3.51 dB (-38%)
Ehang 184: Optimal solutions	214 s (-41%)	3.77 kWh (-71%)	3.62 dB (-36%)

Nevertheless, the convergence behaviour differs for the objectives. The convergence for the objective shortest flight time occurs early, the highest changes of the best solutions for this objective occur during the first generations and stagnate in the later generations. This is an indicator for a premature convergence, which can be caused by low diversity in the population [10]. On the other hand, the convergence of the objective noise addition does not seem to be converged, especially for the Ehang 184 (Fig.10). Also, the objectives shortest flight time and lowest energy consumption share similarities in their convergence that the lowest noise addition does not show.

The two observations of the relationships between the objective functions and the difference convergence behaviours are important to further understand the optimization process for 3D routing with multiple objectives. Explanations for these observations help to further improve the results of 3D route optimization for multiple objectives with evolutionary algorithms and geographical object representations. One explanation for the observation of the relationship between the objective functions relates to the components of the three objective functions. The component of the flight distance appears in the computation of both the flight time and energy consumption: in general, higher flight distances cause worse fitness values in energy and flight time together. Whereas the only distance in the objective function of the least added noise is the distance to the ground, which opposes short flight times and low energy consumptions because the eVTOL is required to travel a further distance. This also explains the discussed negative and low correlations during the convergence process. While the objectives for the energy consumption and the flight time decrease in average, the outcome for the average added noise increases. Furthermore, the component flight distance in the objective function explains, why the optimal solutions for the lowest energy and the shortest flight time are neighbour points in the radial visualization for the aircraft type Ehang 184, but not for the Lilium Jet. The component of the total flight distance is more important for the computation for multicoptor eVTOL aircraft, because the component of the flight speed does not affect the energy consumption in multicoptors. On the other hand, the Lilium Jet can produce lift in higher flight speeds, which decreases energy consumption. Therefore, the energy consumption in the hovering, acceleration and deceleration is much higher than in cruise mode [27]. That can possibly result in cases, where less energy is consumed in longer flight routes. This does not apply for the the flight characteristics of multicoptor aircraft. The flight distance is therefore a common key factor for both aircraft types, but plays a bigger role for multicoptor aircraft.

The second explanations is related to the difficulty to converge to an optimum for an objective. A different level of difficulty can explain the different convergence behaviours of the three objectives shortest flight time, lowest energy consumption and lowest added noise. This relates to the different impacts of changing the genes of an individual on the fitness values. In our study case, a changing gene is a changing 3D point position. An examplary flight path, where only one point position can be changed, as illustrated in Fig. 13, shows the different hardness of searching a better point position. The yellow marked point is the only point that can change position in that simplified scenario. For a better understanding and easier illustration, we remove the height dimension from the objective functions and only consider two-dimensional position changes. For the energy and flight time, at least in this scenario, the positions that improve the fitness are relatively easy to find. If the point gets moved towards the green marked area, the fitness improves, because the course of the flight route straightens. If the point was positioned towards the other direction, the fitness values decrease. In this simplified scenario and for the objectives flight time and energy consumption, the optimal positions for the point are on the line between the two points.

On the other hand, the optimal solution for the minimal added noise, is more difficult to find. The objective function of the lowest added noise is also simplified for this example. The altitude is neglected and the goal is to find the point position in, so that the connecting line lies on noisiest areas. This optimal point position is harder to find because of two reasons. The first reason is, that the space between the noisy areas shows no tendency, at what direction the noisy areas are, most of the values are equal. The second difficulty is, that the closer noisy area, which might be easier to

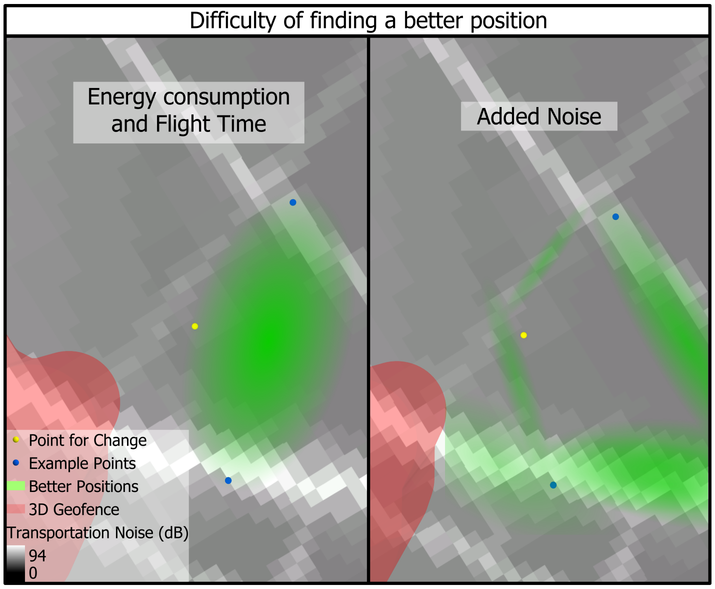
Difficulty of finding a better position
Energy consumption and Flight Time
Added Noise
Point for Change
Example Points
Better Positions
3D Geofence
Transportation Noise (dB)
94
0

Fig. 13: Different hardness of finding the optimum position

find, is not the optimal solution. The best position for the highest noise is the corner at the bottom right of Fig. 13. The best approach to get to this solution would be a totally random positioning, so the solution can only be found by chance. The two different search difficulties of finding a better position in this very simplified scenario resemble the smooth uni-modal versus a rugged multi-modal fitness landscape described in optimization literature [47]. The principle of a fitness landscape is to relate the genes of an individual, in this case the 3D point positions, to its fitness values. An uni-modal fitness landscape describes an optimization problem, where only one global optimum exists, and in a multi-modal fitness landscapes multiple optima exist [47]. Problems with multi-modal fitness landscapes are harder to solve [48]. Even if the problem and the objective functions are highly simplified in this example, the described search processes are one included factor in the more complex objective functions of the 3D route optimization. Even in the more complex objective functions, straight parts of the routes usually lead to a shorter flight time and lower energy consumption. But finding the solutions for minimal added noise is harder and more repetitions are needed. The illustrated different difficulties of finding a better solution explain, why the objective noise converges slower. It converges slower and more erratic (Fig. 10), because the problem is harder to solve. An adaptation for making the search easier to find the noisy areas can be done by smoothing the input noise data. Filtering processes, for example a Gaussian smoothing [49], can cause the raster file to produce a more continuous search space. The loss in detail, on the other hand, would produce a less accurate computation of the added noise at ground. A further examination of preprocessing the noise raster file with filters is part of possible future work.

The observed early convergence of the objective shortest flight time also requires adaptations, as the optimization possibly converged to a local optimum instead of the global optimum [48]. Premature convergence can be caused by a low diversity in the population [10]. One diversity metric can be expressed by the diversity in the objective space with its entropy, that expresses how diverse the fitness values are across the population. Another measurement of the diversity is the entropy of the genes. That expresses, how diverse the gene combination of the individuals in the population are. The illustrated maps in Fig. 11 and Fig. 12 indicate, that many route segments are shared by the different non-dominated solutions. This indicates a low diversity in the population's gene pool in the end of the optimization. The proposed optimization method NSGA II uses a kind of elitism, which is one cause of lower diversity [10]. Also the seeding procedure for the initialization and the small population limit diversity. Especially the initialization with the seeding procedure, along the least distance cost path, produces solutions, that are already near to the shortest distance path. This can explain the earlier convergence of the flight speed and energy consumption, because the component flight distance is one main factor for these objectives. Therefore, in possible future work the population's diversity should be measured [50]. With the known development of the diversity, strategies like self-adapting mutation and crossover probabilities can be applied. For example higher mutation rates are forced in phases of the optimization, where the diversity decreased to a certain level [50]. In order to measure the diversity of the gene pool, the genotypic diversity, methods are required to be identified to effectively measure entropy in 3D flight routes with its specific representation [51].

Next steps of the optimization and also the previous step of the parameter search include an investigation of more runs with other optimization set-ups. The possible parameter values can be analysed in order to measure the impact on the quality of the flight routes and to measure the robustness of the optimization. In order to be able to derive insights about the robustness and the parameter sensitivity, more repetitions are needed. The discussed results and the withdrawn conclusions are possibly from results that were produced by chance. More optimization runs can eliminate those doubts. With additional results, an uncertainty analysis can be done to quantify the uncertainty of the parameter choice [52]. Also the sensitivity of the results to the input parameters could be identified[53]. The sensitivity and uncertainty analysis are part of the possible future work.

Another task for the future is to repeat the optimization for a path in an area where the route can have more variations. In Fig. 11 and Fig. 12 can be observed, that the structure of the geofences do not allow much variation. All routes are positioned equally towards the neighbouring geofences, there is no flight route that surrounds a geofence on the other site. Therefore, the optimization should be repeated in different parts of the study area, where many scattered geofences are located between the start and the ending point. The results from that optimization would be helpful to clarify, whether more diverse flight routes can be found.

4.2 Discussion of the proposed 3D flight route optimization

The objective functions are approximations of the real flight time, energy consumption and added noise. For the objective least noise addition, only the added noise directly underneath the aircraft's position was measured, instead of additionally measuring the added noise of the neighbouring areas. Furthermore, the impact of wind was not considered. Depending on the wind direction and wind speed, the wind conditions can change rapidly in time and can be distributed unevenly in the city area [54]. High wind speeds from specific directions can even lead to dangerous areas, which would need to be modelled in temporal geofences for the optimization [55][3]. Other wind conditions, which are not considered as a hazardous, also have an impact on the flight time, on the energy consumption or the noise addition. The wind can decrease or increase the flight speed and energy consumption, and also refract the sound [56]. The incorporation of the component wind is complex, because it is dynamically changing the search space. As a first step for possible future work, the incorporation of extreme wind conditions is therefore proposed.

Another approximation that affects the objective function lowest energy consumptionc is the weight of the loaded aircraft. It is interesting to see in future work, how sensitive the energy consumption is to the load of the aircraft. Furthermore, it is not clear how noisy the aircraft are in reality, because the air taxi manufacturers did not supply any informations about the sound pressures nor about the sound frequencies. This is a limitation that can not be solved without the cooperation of the manufacturers.

In addition to the considerations regarding the optimization, also aspects of flight management need to be taken into account. If the flight management was not considered, the optimization results are not usable. One important aspect for the flight management is the representation of the flight route and how detailed they are planned. The result of the optimization is a 3D line with an accuracy of less than a centimetre. But in reality, the aircraft are not able to follow the line with that precision, even if the maximum speeds are preserved. Wind gusts or small piloting errors, just as two examples, can lead to an misallocation of the aircraft from the flight route. Therefore, the resulting representation for the real planning would rather be a 3D pipe than a line, by adding three different boundaries around the line, that can be equivalent to a geofence safety boundary management developed by NASA [57]:

- Alert and Awareness Boundary
- Advisory Boundary
- Intervention Boundary

These boundaries correspond to different actions that are initiated by traversing a safety boundary. A crossing of the first boundary initiates a warning, a crossing of the second boundary initiates an alert with precise instructions and the last crossing triggers an external reaction to terminate or re-route a flight [3]. The concept proposed by NASA was planned for geofence boundaries, but the operating Air Taxi companies can use it for an own intervention system, if the flight route lines can not be followed. The distance between the resulting flight line from the optimization to the safety boundaries would depend on the maximum allowed flight speed and the distance to the surrounding geofences [3]. Further informations by the operators, of how detailed the flight route planning will be, are not published yet or not available to the authors. The flight management, including the exact actions while passing the different safety boundaries is important future work. That work would also need to include a simulation with many aircraft at the same time. The flights in the flight route pipes with speed and aircraft size influencing safety distances between operating aircraft need to be planned and simulated [3]. Otherwise the air traffic will be prone to congestions.

Another implication from the lowest added noise results is, that the flight routes can not be static. The goal is, that the yearly average background noise change should not increase by more than 1 dB [36]. This increase of 1 dB is the smallest change that can be detected by a person. The noise levels from the flight routes show, that the added noise is in average $\approx 4dB$. That means, that the flight routes need to be readjusted during the day. It is even possible, that the level of noise increase by only 1 dB might not be achievable at all, or only with less noisy aircrafts than assumed in this work. It is also advisable, that the objective noise becomes more important during night time. One task for future work is also to include the land use, so that the residential areas can be avoided during night times.

4.3 Discussion of the general 3D routing method

The proposed methodology is a new approach for 3D route optimization, and even if the solutions do converge to optima, one main limitation exists in this approach. This limitation is the very high computational effort needed, which is mainly caused by the chosen representation. The intertwined representations and the necessity of updating all three representations after modifications of one is one part. The representation modifications are not the only complex computation regarding the run-time. Reordering the 3D points for example requires a former Near Point Analysis for each point to all other points. This increases the runtime exponentially with the number of 3D points. After the Near Point Analysis, most of the ordering algorithms' run times also increases exponentially with an exponent of 2 and the number of elements as the basis [58]. The Near Analysis, reordering and the representation updates need to be done for each crossover and after every mutation. Another computational complex algorithms are run for the initialization and after every adaptation of 3D flight points. This computation is the value extraction of the minimum flight height and for the background noise at every point position. In addition to that, line intersections with the 3D geofences are necessary operations after every modification of a representation. The mentioned computations are time-consuming, but the computation time also increases linearly with every generation, as illustrated in Fig. 14. One reason can be the variable length of the individual representations, which allows the number of 3D points to increase and leads to more required computations. Another possible but unlikely explanation for this unwanted behaviour is possibly the increasing number of spatial layers stored in the geodatabase, but further information could not be gathered yet.

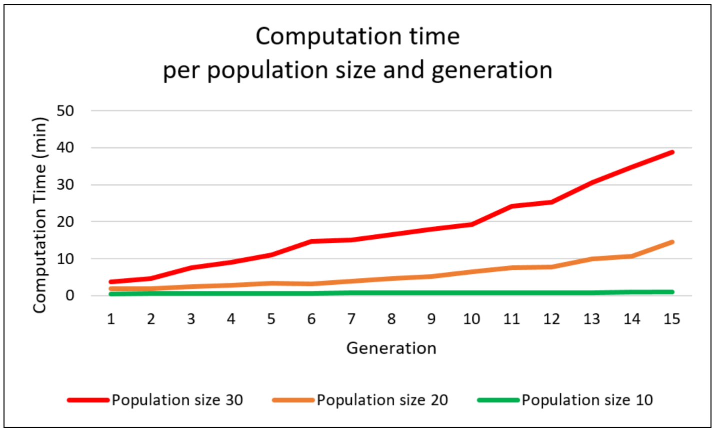

Fig. 14: Computation times per population size and generation

Further and in-depth analysis of each execution step of the run-time and the memory capacity is required to research, which execution tasks of the optimization consume the most time and memory. Also, new research proposes the use of GPU-accelerated spatial databases. These can improve the performance of spatial operations, for example intersections, by a factor of more than 3000 [59].

In addition to using a different technology for improving the processing time, the structure of the algorithm can be altered, too. The concept of co-evolutionary algorithms can improve the performance, which was already successfully applied to another 3D routing problem [13]. The concept divides the whole population into sub populations, which can be distributed in different computation environments and be executed simultaneously. The sub populations can converge simultaneously and then the different sub populations are recombined. The observed increasing computation time in Fig. 14 with increasing population size can be avoided in this way. This strategy is not only faster, the usage of sub population also increases diversity [10].

Another consideration is to change the optimization strategy and use genetic programming. This strategy initializes and optimizes a population of programs [10]. Genetic programming uses programs, usually in tree structures and and these programs are manipulated with evolutionary strategies. This technique can be used for programs, that represent mathematical formulas for high order polynomials, as used by Yeun et al. [60] for surface approximations. This technique can be adapted to create polynomials that represent 3D lines. These lines can then be used for the validation, if they interfere with geofences. This method would not require any reordering or Near Analysis and therefore save computation time. Another expected benefit is a more globally oriented exploration of the search space, because little adaptations in a polynomial can have a big impact on the flight route course. The implementation of other search techniques are also work for the future, for example gradient based trajectory optimization [61] or reinforcement learning [62]. These search techniques show different search space exploration behaviours. They can be used for benchmarking or even for improving the existing flight routes from this work with a local search approach after the solutions from the evolutionary algorithm converged to an optimum.

The reproducibility of this work is currently limited by the dependencies of the GIS software ArcGIS Pro[8] including several license subscriptions. Therefore, a necessary future work is to convert the GIS related functions to Open Source GIS, for example using the libraries of QGIS[9] and the spatial database management from PostgreSQL[10] with the spatial extension PostGIS[11].

Lastly, the generality of the proposed methods is discussed. The study case of the example path in Manhattan in New York was one example of many possible routes from the existing least cost network for the seeding procedure. Therefore, for any different path in the same study area, only the input path needs to be changed for optimizing it. For the application of 3D flight route optimization in a different study area, the method is general applicable if the data requirements are met. The restricted flight areas need to exist as well as the surface with the minimal flight height. An existing explanation [6] and workflow[12] of that calculation are available. The software ArcGIS Pro is required and the described input data needs to be downloaded. After computing the required data, the flight path for the seeding process need to be defined that shall be optimized. If the flight route optimization is intended to be applied for different aircrafts, the only required adaptation is to alter the aircraft specifications. Furthermore, the same method can be used for other kinds of 3D routing, for example 3D routing for submarines. The restricted 3D areas will need to be defined and the objective functions need to be adapted to suite the objectives for the specific 3D routing. For submarines, the travel time and energy consumption will probably be of interest as well, but the physical laws for underwater travelling and therefore the objective functions differ substantially.

5 Conclusion

The air taxi industry is in need of routes that are not interfering with restricted flight areas and are optimized in multiple criteria. In addition to the required flight time and the energy consumption, the criterion of the least added noise is of importance and can determine the quality of a flight route. For finding optimal flight routes, the multi-objective optimization technique NSGA II was used.

The proposed 3D route optimization was applied in Manhattan in New York for the two different eVTOL Lilium Jet and Ehang 184 as representatives of the eVTOL types “Vectored Thrust” and “Multicoptor”. They do have different flight characteristics. The results of the optimization are several non-dominated solutions forming a Pareto front.

Due to the high probability of flight routes intersecting with the restricted air space, additional procedures like seeding and repair methods were introduced. The

[8] https://www.esri.com/de-de/arcgis/products/arcgis-pro/resources

[9] https://www.qgis.org/de/site/

[10] https://www.postgresql.org/

[11] https://postgis.net/

[12] https://github.com/mohildemann/Urban-Air-Mobility-Routing

initialization of individuals with prior information, called seeding, was used to generate valid solutions close to an already existing least distance cost path. The repair methods reposition the 3D points and 3D lines, in case of flight routes intersecting with the restricted air space. Furthermore, an adaptation of an existing 3D point mutation was proposed, which repositions 3D points along a sine curve instead of repositioning single points.

The results proved, that the flight routes represented by geographical point and line objects can be optimized with Evolutionary Algorithms for multiple criteria. Furthermore, the results illustrated that the different aircraft types had an impact on the optimized flight routes. Lastly, the results were used to explain how and why the convergence behaviours differed for the three objectives. The following steps include the summarized observations, how the observations were analysed and interpreted. Compared to the least distance cost path derived with GIS methods only, the optimization produced significantly better results for all criteria. The lowest improvement compared to the least cost path was a noise reduction by 36% with the Ehang jet, the highest improvement was a energy consumption reduction for the eVTOL Lilium Jet by 90%. This leads on to the different convergence behaviours for the three objectives. While the solutions converged early to an optimum for the objective flight time, the convergence towards the objective lowest added noise was slower. The convergence development of the best and average solution suggested, that the process of finding an optimum was still ongoing when the optimization terminated. We supposed that the earlier convergence towards the flight time and energy consumption had two reasons. The first reason is the shared component of the flight route distance, which is one main element influencing both objective functions. Shorter distances lead generally to shorter flight times and lower energy consumption. On the other hand, higher flight positions and longer distances following noisy areas in the city lead subsequently to lower added noises, which is contradictory. The seeding produced flight routes near to the least distance cost path, which increases the probability of producing solutions with shorter flight times and lower energy consumptions, which is not the case for the noise addition. Another reason regards the hardness of finding the optimum. It is easier to find better solutions for lower energy consumption and shorter flight times, because less local optima exists and the fitness landscape is smoother with less local optima. The background noise has many local highs and multiple large areas with same noise levels, which increases the search difficulty for better solutions and slows down the convergence. It also explains the observed more erratic convergence behaviour.

The results showed that the aircraft type with its different flight characteristics matters for finding optimal flight routes. The biggest differences were observed in the objective energy consumption between the Ehang 184 and the Lilium Jet, which is caused by the aircraft type. The Lilim Jets' wings produce lift in higher flight speeds, which is not the case for the multicoptor aircraft from Ehang.

The approach of optimizing 3D routes consisting of geographical 3D objects has one main drawback, which is the computational complexity. High computation times limited a more intensive parameter search, bigger population sizes and more generations to further converge to better compromise solutions. The identified main reasons for the high computation times are the updates after modification of any of the three intertwined representations and the repair method for invalid solutions. This leads to the conclusion, that either the used technology or the evolutionary strategy needs to be adapted. A different technology like GPU-accelerated spatial databases or co-evolutionary algorithms could accelerate the computation with the same intertwined presentation type. On the other hand, the use of different search techniques like genetic programming can eliminate the need of the 3D point presentation by its different exploration technique of the search space.

In summary, the proposed methodology for optimizing 3D flight routes has been proved to be successful. The proposed representation of individuals and the evolutionary operators are expandable, but allowed to produce valid compromise solutions for all criteria. The results can be directly used for further considerations for the flight management. The most important conclusion for the urban air management is the resulting average noise additions of more than 3 *dB*, which is considered as an unacceptable long-term annoyance. This implies the need of temporally changing flight route courses if the noise pressure of the eVTOLs can not be reduced significantly. The proposed method is also applicable for other urban areas, different aircraft types and other 3D routing applications. Furthermore, the method can be applied to any 3D routing in an 3D environment with multiple constraints and multiple objectives.

Appendix

NSGA II

The general process of the NSGA II is explained briefly. At first, the individuals/solutions of the population need to be evaluated for all criteria. In a next step, the ranking, all non-dominated solutions are derived. By adding the non-dominated solutions to the first pareto front with the non-domination rank 1, the step is repeated for the remaining solutions until all solutions are assigned to a pareto front with the corresponding rank of each pareto front. The solutions belonging being assigned to the second rank are the non-dominated solutions if the solutions of the first rank were ignored. The rank is used for the selection process of the individuals, where the solutions are compared to each other. If two solutions have the same non-domination rank, another metric is used to choose the superior solution: The crowding distance. This metric estimates the density of the region around the solution within the objective value space. This means, solutions whose objective values are similar to other solutions, get assigned a lower crowding distance than solutions whose objective values are more distinct. If the non-domination rank is equal, solutions witha higher crowding distance are preferred. This selection therefore prefers solutions with a bigger diversity in their phenotype. After producing and evaluating the offsprings in every generation, the non-domination rank and the crowding distance are used again. The standard approach of evolutionary algorithm is the replacement of the parent generation with the offspring generation. In the NSGA II, the individuals that shall form the population of the next generation are selected from the parent and offspring population together. Until the population size is reached, the solution with the best ranks are selected. In the case, that many solutions have the same rank for too few open slots, the crowding distance determines the selection. Solutions with a higher crowding-distance are solutions that have a higher crowding distance [14]. These solutions are preferred.

References

1. X. Sun, S. Wandelt, and E. Stumpf. Competitiveness of on-demand air taxis regarding door-to-door travel time: A race through europe. *Transportation Research Part E: Logistics and Transportation Review*, 119:1–18, 2018.
2. R. Goyal. Urban air mobility (uam) market study: Presented to: National aeronautics and space administration - aeronautics research mission directorate. https://ntrs.nasa.gov/search.jsp?r=20190001472, 2018.
3. D. Geister. Integrating uas into the future aviation system: A flexible approach enabling large-scale uas operations: German aerospace center (dlr). *Institute of Flight Guidance*, 2017.
4. S. Baur, S. Schickram, A. Homolenko, N. Martinez, and A. Dyskin. Urban air mobility: The rise of a new mode of transportation, 2018.
5. National Transportation Noise Map. Exposure to noise from aviation and interstate highways. http://osav.usdot.opendata.arcgis.com/, 2014.
6. M. Hildemann and C. Delgado. An adaptable and scalable least cost network for air-taxis in urban areas study area: Manhattan, new york. In *Accepted Short Papers and Posters from the 22nd AGILE Conference on Geo-information Science*. AGILE, 2019.
7. A. Durmaz, E. Ünal, and C. Aydın. Automatic pipeline route design with multi-criteria evaluation based on least-cost path analysis and line-based cartographic simplification: A case study of the mus project in turkey. *ISPRS International Journal of Geo-Information*, 8(4):173, 2019.
8. U. Atila, Ismail R. Karas, and A. Rahman. A 3d-gis implementation for realizing 3d network analysis and routing simulation for evacuation purpose. In J. Pouliot, S. Daniel, F. Hubert, and A. Zamyadi, editors, *Progress and New Trends in 3D Geoinformation Sciences*, volume 29 of *Lecture Notes in Geoinformation and Cartography*, pages 249–260. Springer Berlin Heidelberg, Berlin, Heidelberg, 2013.
9. S. Sandurkar and W. Chen. Gaprus - genetic algorithms based pipe routing using tessellated objects. *Computers in Industry*, 38(3):209–223, 1999.
10. S. Sivanandam and S. Deepa. *Introduction to genetic algorithms*. Springer, Berlin, 2008.
11. R. G. Newell and T. L. Sancha. The difference between cad and gis. *Computer-Aided Design*, 22(3):131–135, 1990.
12. A. Amirjanov and F. Sadikoglu. Linear adjustment of a search space in genetic algorithm. *Procedia Computer Science*, 120:953–960, 2017.
13. Z. Peng, J. Wu, and J. Chen. Three-dimensional multi-constraint route planning of unmanned aerial vehicle low-altitude penetration based on coevolutionary multi-agent genetic algorithm. *Journal of Central South University of Technology*, 18(5):1502–1508, 2011.
14. K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan. A fast and elitist multiobjective genetic algorithm: Nsga-ii. *IEEE Transactions on Evolutionary Computation*, 6(2):182–197, 2002.
15. J. H. Holland. *Adaptation in natural and artificial systems: An introductory analysis with applications to biology, control, and artificial intelligence*. University of Michigan Press, Ann Arbor, Michigan, 1975.
16. Lilium Aviation. Mission: Aircraft for everyone. https://lilium.com/mission/, 2019.
17. Federal Aviation Administration. Integration of civil unmanned aircraft systems(uas) in the national airspace system (nas) roadmap, 2019: First edition.
18. Federal Aviation Administration. *Pilot's Handbook of Aeronautical Knowledge*. Skyhorse Publishing Company Incorporated and Thomas Allen & Son Limited [Distributor], New York and Markham, 2017.
19. OpenStreetMap contributors. https://planet.osm.org, 2019.
20. Flight Obstacles. Digital obstacle file, https://www.faa.gov/air_traffic/flight_info/aeronav/digital_products/dof/, 04.11.2019.
21. Open Data NYC. Open data of new york city: New york city population by neighborhood tabulation areas. https://opendata.cityofnewyork.us/, 2018.
22. S. Shekhar and H. Xiong, editors. *Encyclopedia of GIS*. Springer US, Boston, MA, 2008.
23. Esri. Real-time data feeds and analytics: Managing geofences. http://enterprise.arcgis.com/de/geoevent/latest/administer/managing-geofences.htm, 2018.
24. T. Friedrich and M. Wagner. Seeding the initial population of multi-objective evolutionary algorithms: A computational study. *Applied Soft Computing*, 33:223–230, 2015.

25. Electric VTOL News. evtol classifications. https://evtol.news/classifications/, 2019.
26. EHANG. Ehang 184 specifications. http://www.ehang.com/ehang184/specs/, 2019.
27. A. Bacchini and E. Cestino. Electric vtol configurations comparison. *Aerospace*, 6(3):26, 2019.
28. C. Y. Lee and Antonsson E. K., editors. *Dynamic Partitional Clustering Using Evolution Strategies*, IEEE, 2000.
29. V. Andrievskii and H. Blatt. *Discrepancy of Signed Measures and Polynomial Approximation*. Springer New York, New York, NY, 2002.
30. P. Koch, S. Bagheri, W. Konen, C. Foussette, P. Krause, and T. Bäck. A new repair method for constrained optimization. In A. Esparcia-Alcázar, S Silva, and J. Jiménez-Laredo, editors, *Proceedings of the 2015 on Genetic and Evolutionary Computation Conference - GECCO '15*, pages 273–280, New York, New York, USA, 2015.
31. E. Torenbeek and H. Wittenberg. *Flight Physics*. Springer Netherlands, Dordrecht, 2009.
32. M. Sadraey. *Aircraft performance analysis*. VDM Verlag Dr. Müller, Saarbrücken, Germany, 2009.
33. U.S. Environmental Protection Agency. Office of noise abatement and control. http://www.nonoise.org/library/levels74/levels74.htm: Epa, 1974.
34. World Health Organization. Guidelines for community noise: https://www.nh.gov/osi/energy/programs/documents/sb99-who-guidelines-community-noise.pdf. 1999.
35. L. E. Kinsler. *Fundamentals of acoustics*. Wiley, New York and Chichester, 4 edition, 2000.
36. Uber Elevate. Fast-forwarding to a future of on-demand urban air transportation: White paper. https://www.uber.com/us/en/elevate/, 2016.
37. M. J. Reddy and D. N. Kumar. Elitist-mutated multi-objective particle swarm optimization for engineering design. In D. B. .A. Khosrow-Pour, editor, *Encyclopedia of Information Science and Technology, Third Edition*, pages 3534–3545. IGI Global, 2015.
38. K. A. de Jong and W. M. Spears. A formal analysis of the role of multi-point crossover in genetic algorithms. *Annals of Mathematics and Artificial Intelligence*, 5(1):1–26, 1992.
39. K. Deb and A. Kumar. Interactive evolutionary multi-objective optimization and decision-making using reference direction method. In H. Lipson, editor, *Proceedings of the 9th annual conference on Genetic and evolutionary computation - GECCO '07*, page 781, New York, USA, 2007.
40. P. Hoffman, G. Grinstein, K. Marx, I. Grosse, and E. Stanley. Dna visual and analytic data mining. In *Proceedings. Visualization '97 (Cat. No. 97CB36155)*, pages 437–441. IEEE, 19-24 Oct. 1997.
41. P. J. Korhonen and J. Laakso. A visual interactive method for solving the multiple criteria problem. *European Journal of Operational Research*, 24(2):277–287, 1986.
42. X. Ma, Q. Zhang, G. Tian, J. Yang, and Z. Zhu. On tchebycheff decomposition approaches for multiobjective evolutionary optimization. *IEEE Transactions on Evolutionary Computation*, 22(2):226–244, 2018.
43. R. T. Marler and J. S. Arora. The weighted sum method for multi-objective optimization: new insights. *Structural and Multidisciplinary Optimization*, 41(6):853–862, 2010.
44. A Hassanat, K. Almohammadi, E. Alkafaween, E. Abunawas, A. Hammouri, and S. Prasath. Choosing mutation and crossover ratios for genetic algorithms: A review with a new dynamic approach. *Information*, 10(12):390, 2019.
45. J. Schaffer, Rich Caruana, Larry Eshelman, and Rajarshi Das, editors. *A Study of Control Parameters Affecting Online Performance of Genetic Algorithms for Function Optimization*, George Mason University, 1989.
46. K. Deb and S. Chaudhuri. I-mode: An interactive multi-objective optimization and decision-making using evolutionary methods. In Shigeru Obayashi, Kalyanmoy Deb, Carlo Poloni, Tomoyuki Hiroyasu, and Tadahiko Murata, editors, *Evolutionary Multi-Criterion Optimization*, volume 4403 of *Lecture Notes in Computer Science*, pages 788–802. Springer Berlin Heidelberg, Berlin, Heidelberg, 2007.
47. van Cleve, J. Weissman, D. B. Measuring ruggedness in fitness landscapes. *Proceedings of the National Academy of Sciences of the United States of America*, 112(24):7345–7346, 2015.
48. D. Whitley, M. Vose, J. Horn, and D. Goldberg. Genetic algorithm difficulty and the modality of fitness landscapes. *Illinois Genetic Algorithms Laboratory*, 1995.

49. P. Longley, M. F. Goodchild, D. Maguire, and D. W. Rhind. *Geographic information science & systems*. John Wiley & Sons, Hoboken, New Jersey, fourth edition edition, 2015.
50. P. Kulvanit, T. Piroonratana, N.l Chaiyaratana, and D. Laowattana. Evolutionary multi-objective optimisation by diversity control. In D. Hutchison, T. Kanade, J. Kittler, J. M. Kleinberg, F. Mattern, J. C. Mitchell, M.i Naor, O. Nierstrasz, C. Pandu R., B. Steffen, M. Sudan, D. Terzopoulos, D. Tygar, M. Y. Vardi, G. Weikum, D. Grigoriev, J. Harrison, and E. A. Hirsch, editors, *Computer Science – Theory and Applications*, volume 3967 of *Lecture Notes in Computer Science*, pages 447–456. Springer Berlin Heidelberg, Berlin, Heidelberg, 2006.
51. G. Corriveau, R. Guilbault, A. Tahan, and R. Sabourin. Review and study of genotypic diversity measures for real-coded representations. *IEEE Transactions on Evolutionary Computation*, 16(5):695–710, 2012.
52. M. Li and S. Azarm. Multiobjective collaborative robust optimization with interval uncertainty and interdisciplinary uncertainty propagation. *Structural and Multidisciplinary Optimization*, 130(8):6, 2008.
53. S. M. Blower and H. Dowlatabadi. Sensitivity and uncertainty analysis of complex models of disease transmission: An hiv model, as an example. *International Statistical Review / Revue Internationale de Statistique*, 62(2):229, 1994.
54. R. D. Bornstein and D. S. Johnson. Urban-rural wind velocity differences. *Atmospheric Environment (1967)*, 11(7):597–604, 1977.
55. M. N. Stevens and E. M. Atkins. Generating airspace geofence boundary layers in wind. *Journal of Aerospace Information Systems*, pages 1–12, 2019.
56. A. R. Kriebel. Refraction and attenuation of sound by wind and thermal profiles over a ground plane. *The Journal of the Acoustical Society of America*, 51(1A):19–23, 1972.
57. E. T. Dill, S. D. Young, and K. J. Hayhurst. Safeguard: An assured safety net technology for uas. In *2016 IEEE/AIAA 35th Digital Avionics Systems Conference (DASC)*, pages 1–10. IEEE, 25/09/2016 - 29/09/2016.
58. K. Alkharabsheh, I. Alturani, and N. Zanoon. Review on sorting algorithms a comparative study. *International Journal of Computer Science and Security (IJCSS)*, 7, 2013.
59. L. C. Villa Real and B. Silva. Full speed ahead: 3d spatial database acceleration with gpus. 2018.
60. Y. S. Yeun, Y. S. Yang, W. S. Ruy, and B. J. Kim. Polynomial genetic programming for response surface modeling part 1: a methodology. *Structural and Multidisciplinary Optimization*, 29(1):19–34, 2005.
61. S. Chauhan and J. Martins. Tilt-wing evtol takeoff trajectory optimization. *Journal of Aircraft*, 57(1):93–112, 2020.
62. K. Ota, D. Jha, T. Oiki, M. Miura, T. Nammoto, D. Nikovski, and T. Mariyama. Trajectory optimization for unknown constrained systems using reinforcement learning. Machine Learning, 2019.