The Aeronautical Journal (2023), **127**, pp. 2125–2142
doi:10.1017/aer.2023.68

REGULAR PAPER

Prioritising paths: An improved cost function for local path planning for UAV in medical applications

A. Thoma[1,2], K. Thomessen[1], A. Gardi[2,3], A. Fisher[2] and C. Braun[1]

[1]Department of Aerospace Engineering, FH Aachen, Postfach 10 05 60, Aachen, Germany, [2]School of Engineering, Royal Melbourne Institute of Technology, GPO Box 2476V, Melbourne, Victoria, 3001, Australia and [3]Department of Aerospace Engineering, Khalifa University, PO Box 127788, Abu Dhabi, UAE
Corresponding author: A. Thoma; Email: a.thoma@fh-aachen.de

Received: 28 April 2023; **Revised:** 27 June 2023; **Accepted:** 5 July 2023

Keywords: Path planning; Cost function; Multi-objective optimization

Abstract
Even the shortest flight through unknown, cluttered environments requires reliable local path planning algorithms to avoid unforeseen obstacles. The algorithm must evaluate alternative flight paths and identify the best path if an obstacle blocks its way. Commonly, weighted sums are used here. This work shows that weighted Chebyshev distances and factorial achievement scalarising functions are suitable alternatives to weighted sums if combined with the 3DVFH* local path planning algorithm. Both methods considerably reduce the failure probability of simulated flights in various environments. The standard 3DVFH* uses a weighted sum and has a failure probability of 50% in the test environments. A factorial achievement scalarising function, which minimises the worst combination of two out of four objective functions, reaches a failure probability of 26%; A weighted Chebyshev distance, which optimises the worst objective, has a failure probability of 30%. These results show promise for further enhancements and to support broader applicability.

Nomenclature

A	atainable set
BVLoS	beyond visual line-of-sight
D	drag force
DoE	design of experiments
$E_{kin,i}$	kinetic energy in timestep i
$E_{pot,i}$	potential energy in timestep i
$E_{thrust,i}$	thrust energy in timestep i
E_{tot}	estimated total energy
E_i	estimated energy in timestep i
f	failure probability
F	Feasible domain
FOM	figure of merit
F_p	epsilon constraint function
g	gravitational acceleration
g_j	inequality constraint j
i	time step index, objective index
I_q	subset of N_n of cardinality q
m	number of constraints, mass
n	dimension
n_f	number of failed flights
n_s	number of successful flights

N	natural numbers
o_i	objective function i
o_i^O	utopian point of o_i
$\boldsymbol{O}(\boldsymbol{x})$	set of objective functions o_i
q	degree of flexibility
S_{ref}	reference surface
T	reference thrust
U	minimal solution, minimal point
UAV	unmanned aerial vehicle
v_i	velocity in timestep i
v_x, v_y, v_z	velocity component
VLoS	visual line-of-sight
w_i	weight i
$\mathbf{x}$	vector of the feasible domain
z	flight altitude
3DVFH*	3D vector field histogram *
Δt	size of time step
ϵ	epsilon constraint
ρ	air density

1.0 Introduction

Unmanned aerial vehicles (UAVs) are ideal for various applications, including the medical sector. Government, industry and academia are working on projects like delivering medical supplies, transporting organs and supporting first responders with UAV-based tele-doctors. Due to limited autonomous obstacle avoidance capabilities, most UAVs can only operate within their remote pilot's visual line-of-sight (VLoS) [1, 2]; however, the medical sector, similarly to various other current and emerging application domains, requires UAV operation beyond visual line-of-sSight (BVLoS) [3]. In the long run, the medical sector will require even fully autonomous BVLoS operations to provide nationwide, inexpensive medical coverage. Even though several obstacle avoidance algorithms exist, reliable and efficient evasion of static and moving obstacles is still challenging, especially for small UAVs below 5 kg with low computational power, sensor and battery capacity. Hence, more robust and efficient obstacle avoidance algorithms are required to overcome this challenge [4]. Although obstacle detection has recently progressed, the reliable planning of safe avoidance manoeuvres in the very stringent timeframe between detecting an obstacle and the potential collision remains particularly challenging [5].

Therefore, path planning is a crucial aspect of autonomous BVLoS UAV operation, where the most efficient and feasible path must be found between a given position and a goal point [6]. Two different path planning approaches are available. A global path planner identifies a suitable path between a start point and a goal point based on given information about the environment. However, global path planners cannot react to unknown or unforeseen obstacles [6]. Therefore, a global path planner always needs an accurate and complete *a priori* database and representation of the environment. On the contrary, local path planners find suitable paths between the current position and the goal based on information gathered by a sensor system. Local path planners do not have any additional information about the environment, allowing them to operate in changing environments without additional effort [6]. However, local path planners have a higher risk of failing to find a feasible path from their current position to the goal point than global path planners [7]. Nonetheless, a local path planner is required if a flight mission requires flight at low altitudes in uncontrolled environments, even for a short distance. Notably, medical applications have a variety of unknown components, e.g. terrain, obstacles, exact goal location. At the same time, UAVs capable of autonomous replanning bring several advantages to the medical sector [8]. For instance, the UAV can quickly fly towards the patient as soon as the goal point is known. This goal point might be in an unknown area or an area with unknown obstacles. The goal point might also be located where a safe remotely piloted operation cannot be ensured. This situation requires an efficient

https://doi.org/10.1017/aer.2023.68 Published online by Cambridge University Press

and reliable local path planning algorithm. Unfortunately, currently available local path planning algorithms have insufficient reliability. This work tackles the insufficient reliability of current algorithms by investigating the influence of the cost function type of the local path planning algorithm on its reliability.

Path planning requires finding the most efficient and feasible path from the current position to the goal while considering various constraints such as obstacles, time or distance. This process typically requires evaluating and comparing multiple alternative paths to identify the best one. Therefore, most path planning algorithms generate a set of feasible paths and select the optimal. However, defining which path is the best is nontrivial. Frequently, path planning considers multiple criteria, leading to a multi-objective optimisation problem that requires a multiple-criteria decision strategy. Most local and global path planners use cost functions to reduce the multi-objective optimisation problem to a single-objective optimisation problem and identify the ideal flight path. Traditionally, multi-objective optimisation strategies rely on weighted sums to prioritise the different aspects in an *a priori* articulation of preference [9, 10]. However, they disregard interactions between path planning aspects, are sensitive to normalisation and weighting parameters, and cannot find all Pareto optimal solutions. Only a few works consider alternative cost function types, e.g. Ref. (11) for multi-objective UAS trajectory optimisation.

Therefore, this work investigates multiple alternative types of cost functions. First, we discuss them on a theoretical basis. Besides the classical weighted sum, we investigate variants of weighted products, weighted Chebyshev distance, and factorial achievement scalarising. Second, we test these functions with the 3DVFH* in simulations of multiple short-range flight missions in various environments. The 3DVFH* is a well-known and commonly used local path planning algorithm. It relies on a weighted sum and represents many optimisation-based local path planning algorithms. We also performed a parameter sweep for the weighting factors to identify ideal parameters and determine the function's potential. Then, we compare the performance of the alternative cost functions regarding their safety (crash probability), reliability (mission failure probability) and suitability for medical applications (average flight time, distance and energy consumption). We also investigate the sensitivity of the different cost function types on the choice of weights. We finally present an improved version of the 3DVFH* with higher reliability and better performance.

The contributions of this work summarise as follows:

1. An in-depth analysis and comparative evaluation of the influence of different cost function types for collision avoidance path planning (implemented in 3DVFH*) in terms of their reliability (failure probability) and efficiency (energy consumption).
2. Providing important recommendations for selecting or developing multi-objective optimisation-based local path planners to target safety and efficiency performance in realistic scenarios.

1.1 Multi-objective path planning

Several hundred path planning algorithms exist for 2D, 2.5D and 3D planning and for local and global path planning [12]. Most rely on classic deterministic approaches, often based on an optimisation approach [13]. Several algorithms rely on weighted sums as cost functions [14–17]; however, to the best of the authors' knowledge, no comprehensive investigation of the influence of the type of cost function on the local path planning result is available.

Using artificial neural networks and machine learning also becomes increasingly popular for local path planning tasks [18]. In general, three types of machine learning are used in local path planning. First, imitation learning trains a neural network to replicate a human pilot's actions, e.g. Ref. (19). These neural networks work well in familiar scenarios; however, they struggle in novel situations. Second, reinforcement learning uses a reward function to train a neural network to achieve a specific goal, as in Ref. (20). This approach is well suited to integrate several objectives with an adequate reward function. However, defining the reward function is very difficult [19]. These algorithms also struggle in novel situations. Third, deep neural networks are trained to map a specific reaction to a given input (usually a camera image). This approach proved efficient in very specific situations (e.g. for a UAV

https://doi.org/10.1017/aer.2023.68 Published online by Cambridge University Press

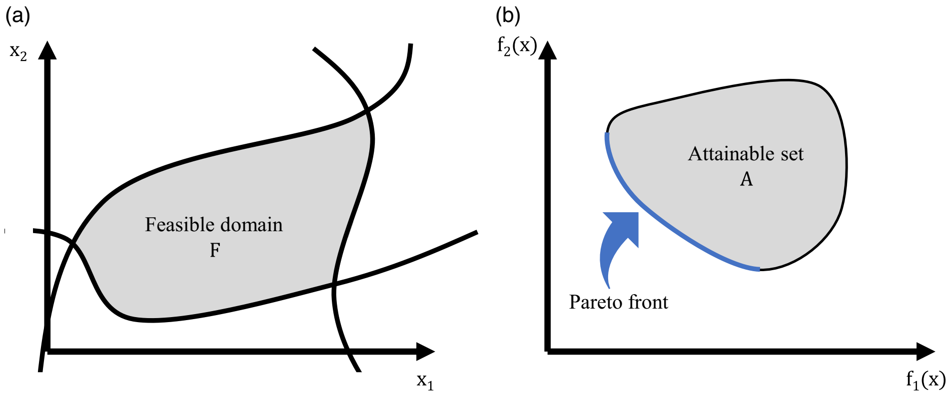

Figure 1. *Feasible domain (a) and attainable set (b) of an arbitrary two-objective optimisation problem.*

flying through corridors [21]); however, this approach also struggles in new situations. It is also worth noting that all machine learning methods are currently not certifiable in line with aeronautical standards and regulations and that considerable challenges are still faced when evaluating their safe adoption in real platforms as the ethics and regulations concerning the use of artificial intelligence in increasingly autonomous vehicles are still being drafted [22]. Some approaches combine classic deterministic local path planning with deep neural networks, e.g. Ref. (13). While these neural network-based algorithms have great potential, classic deterministic local path planning is still more robust and predictable than these novel approaches.

Other works focus on optimised path planning for swarms of UAVs [23, 24] or goal points (commonly known as the travelling salesman problem) [25]. Other approaches consider different types of threads or prioritise different tasks [26]. These algorithms often use meta-heuristics to identify the best solution, e.g. Ref. (27) or optimal control theory to identify the best path [22, 28–30]; however, these approaches are mostly global planners and therefore require significant a priori knowledge of the environment, as discussed, although it is possible to formulate real-time variants.

2.0 Theoretical background and methodology

Most path-planning problems are multi-objective optimisation problems with multiple constraints and typically conflicting objectives in multiple layers. During path planning, goal-driven planning, i.e. flying in the direction of the goal, usually contradicts feasiblemanoeuvres when avoiding obstacles, e.g. flying around an obstacle perpendicular to the goal direction. On a more global scale, energy consumption, flight time, flight distance and risk of failure might conflict with one another. These various objectives can be formulated independently as objective functions o_i with the goal to minimise these functions as exemplarily given in Equation (1).

$$min\left\{\boldsymbol{O}(\boldsymbol{x}) = [o_1(x),\ o_2(x), \ldots, o_n(x)]^T\right\} \tag{1}$$

$$\text{Subject to } g_j(x) \leq 0; j = 1,\ 2,\ \ldots,\ m \tag{2}$$

Here, n denotes the number of objective functions, g_j represents the constraints, and m denotes the number of inequality constraints. The vector of variables $\boldsymbol{x} \in E^n$ gives the feasible domain $\mathbf{F}= \left\{\boldsymbol{x}|g_j(\boldsymbol{x}) \leq 0,\ j = 1,\ 2,\ \ldots,\ m\right\}$. The attainable set is given by $\boldsymbol{A} = \{\boldsymbol{O}(\boldsymbol{x})|\boldsymbol{x} \in \boldsymbol{F}\}$.

However, defining optimal conditions for problems with a vector-valued objective function is generally impossible [31]. Optimising such functions gives set-valued solutions: a Pareto front in the objective function space and a Pareto set in the design variable space. Figure 1(a) illustrates a design variable space

https://doi.org/10.1017/aer.2023.68 Published online by Cambridge University Press

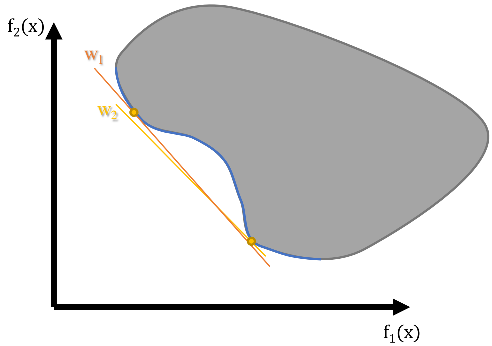

Figure 2. *Objective space with a non-convex region in the Pareto front. w_1 and w_2 belong to two minimally different sets of weights for the weighted sum.*

and its feasible space. Figure 1(b) illustrates an objective function space and its attainable set for a problem with two objectives. The Pareto front is the line of the attainable set of non-dominated points. Points are non-dominated if one objective cannot improve without degrading another. The Pareto set consists of the design variables belonging to the non-dominated points of the Pareto front.

Two main strategies exist to identify the Pareto front: scalarisation and direct solution methods. Scalarisation converts the multi-objective problem into a single-objective problem or series of single-objective problems – in engineering, commonly known as cost functions. Most direct solution methods use populations to test the attainable set, e.g. multi-objective genetic algorithms; however, direct solution methods lack theoretical convergence proprieties, and unless a very high number of points are evaluated, no sufficient knowledge about the Pareto front is obtained.

Identifying a large set of Pareto points is often unnecessary in an application, as ultimately, only one preferred point is needed. Many methods to find one preferred Pareto optimal point use scalarisation. Therefore, the most common methods are presented and discussed in the following.

2.1 Weighted sum

The ideal solution for all presented methods is a minimal U. In the case of the weighted sum, this scalar is calculated as per the following equation:

$$U = \sum_{i=1}^{n} w_i o_i(\boldsymbol{x}) \tag{3}$$

where w_i are the weightings of the objective functions $o_i(\boldsymbol{x})$.

Weighted sums are the most common and intuitive strategy to transform a multi-objective optimisation problem into a single-objective optimisation problem in engineering [32]. Weights assign different importance to the different objectives of the multi-objective problem [33]. In the objective space, a given set of weights corresponds to an iso-cost line. In the case of a bi-objective problem, the weights define the slope of a set of parallel, straight lines in the objective function space. This method finds Pareto optimal points in non-convex regions of a Pareto front by alternating the weights of the weighted sum. However, it fails to find all non-dominated points in a non-convex Pareto front, as illustrated by Fig. 2. Minimal weight changes might lead to fundamentally different Pareto optimal points – one of the most significant weaknesses of the weighted sum approach.

A priori articulation of preferences identifies a single solution that presumably reflects the desired conditions. However, the sensitivity to the choice of weighting parameters, as indicated in Fig. 2, shows the difficulty of this task. Additionally, the magnitude of o_i also defines the influence of various objectives [32]. If the magnitude of o_i varies greatly, normalisation is required. The type of normalisation, however, defines the influences on the overall result and is arbitrary to a certain degree [34]. Various

https://doi.org/10.1017/aer.2023.68 Published online by Cambridge University Press

works investigate methods to set ideal weights, e.g. Ref. (9), or the broader influence of weights on the results, e.g. Ref. (32). However, none fully overcomes the disadvantages. The most common and intuitive engineering approach is associating individual objectives with importance. The weights w_i accordingly represent the importance of the respective objectives. Therefore, the weighted sum focuses on optimising the more important objectives. However, the shape of the attainable set defines how sensitive the result is to the choice of weight. If the shape is non-convex, even minimal changes in weighting factors might lead to completely different solutions (see Fig. 2). Frequently, local path planning algorithms use path objectives (smoothness, obstacle vicinity, goal direction) to identify the optimal path. However, relevant mission objectives (success, distance, time, energy) may change during the flight (e.g. as a function of weather conditions) and are only ultimately known after the flight. Therefore, the weights of the path objectives are chosen such that they hopefully lead to the desired mission outcome. In practice, evidence suggests that the weighted sum might lead to a difficult-to-tune local path planner with unreliable performance because of the dependency of the result on the shape of the attainable set, the magnitude of the objectives and the chosen weights.

2.2 Weighted product

The weighted product is analogue to the weighted sum, but the individual objective functions o_i are multiplied and weighted by an exponential w_i as in:

$$U = \prod_{i=1}^{n} o_i(\boldsymbol{x})^{w_i} \tag{4}$$

Weighted products require $o_i(\boldsymbol{x}) > 0$, $\forall i$. Contrary to weighted sums, weighted products do not require normalisation. However, weighted products also overvalue extremes: if one of the objective functions o_i gives an exceptionally high or low value, the influence on the overall result is strong [33]. Moreover, they may incur saturation of computational precision of the calculator or library used, particularly in the presence of several objectives. Similar to weighted sum, the weighted product focus on optimising the most important objectives.

2.3 Weighted Chebyshev distance

The weighted Chebyshev distance optimises the worst of the objective functions:

$$U = \max\left\{w_i\left[o_i - o_i^0\right]\right\} \tag{5}$$

Here, o_i^0 denotes the utopian point with the best result for every individual objective function; $o_i^O = \min\{o_i(\boldsymbol{x}) | \boldsymbol{x} \in \boldsymbol{X}\}$ (35). The weighted Chebyshev distance aims to find the lowest worst objective relative to the utopian point possible.

2.4 Factorial achievement scalarising function

Achievement scalarising functions are based on natural decision-making and using desirable reference objectives rather than weighting factors [36].

$$U = \max_{I^q \subseteq N_n : |I^q| = q} \left\{\sum_{i \in I^q} \max\left[w_i \cdot \left(o_i(\boldsymbol{x}) - o_i^u\right), 0\right]\right\} \tag{6}$$

In parameterised achievement scalarising functions, as in Equation (6), an integer parameter $q \in N_n$ controls the degree of flexibility from a linear L_1 metric to a Chebyshev L_∞. I^q denotes a subset of N_n of cardinality q [37]. Achievement scalarising functions aim to find the lowest worst combination of objectives relative to the utopian point depending on the degree of flexibility.

https://doi.org/10.1017/aer.2023.68 Published online by Cambridge University Press

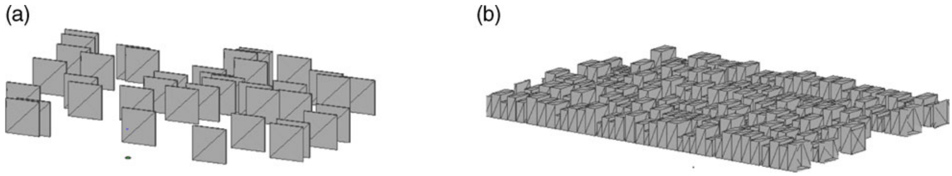

Figure 3. *Examples of two test scenarios. (a) is an environment with flat rectangular obstacles and (b) represents an urban environment with various buildings of different sizes and dimensions.*

2.5 Epsilon-constraint method

$$U = \min\left\{F_p(\boldsymbol{x})\right\} \tag{7}$$

$$\text{Subject to } F_i(\boldsymbol{x}) \leq \epsilon_i, \ \ i = 1, \ldots, n, \ \ i \neq p \tag{8}$$

The Epsilon-constraint method converts all objectives except one, denoted as F_p, into bounds – leading to a single optimisation problem [38]. By systematically varying bounds on the constraints, the Pareto front is generated. Therefore, the epsilon-constraint method does not lead to an aggregated objective function. This approach's advantage is that the parts of a non-convex Pareto front are also identified. However, it is also difficult to choose an appropriate value for epsilon, shifting the problem of defining adequate weights to a problem of defining good epsilon values. The epsilon constraint method is also computationally heavy because it has to solve several optimisation problems at once; therefore, this method is not further investigated.

2.6 Experimental setup

We use the 3DVFH* as a testing algorithm. The 3DVFH* combines a 3D vector field histogram with the A* algorithm. It generates several candidate paths, usually consisting of five waypoints 2m apart, based on the search behaviour of A*. It then evaluates all candidate paths with a cost function. The standard 3DVFH* has a weighted sum as a cost function, as in Equation (9). The 3DVFH* was introduced in Ref. (39).

$$U = w_{yaw} \cdot o_{yaw} + w_{pitch} \cdot o_{pitch} + w_{smooth} \cdot o_{smooth} + o_{obstacle} + o_{heuristics} \tag{9}$$

All o_i in Equation (9) depend on $\boldsymbol{x}$ giving $o_i(\boldsymbol{x})$; however, we omitted the $(\boldsymbol{x})$ in Equation (9) for abbreviation. The total cost U consists of the cost for horizontal deviation from the direct goal direction o_{yaw}, weighted by w_{yaw}, the cost for vertical deviation from the direct goal direction o_{pitch}, weighted by w_{pitch}, the cost for deviation from the previous flight direction o_{smooth} weighted by w_{smooth}, the cost of proximity to obstacles $o_{obstacle}$ and heuristic cost $o_{heuristic}$, that describe the remaining distance to the goal. The algorithm estimates the cost of alternative paths and chooses the cheapest path.

We define two main categories of test scenarios: simple obstacle avoidance and urban environments. We defined 900 different short-distance flight scenarios in which the UAV has to navigate past one to 40 flat obstacles, standing or floating in the air. We also defined 1,408 challenging urban environments to evaluate the performance of the different cost functions in different, realistic situations. The urban environments consist of various cuboidal obstacles in shapes and sizes typical for various buildings. The obstacles stand in rows with different distances between them. Figure 3 shows sample images of both types of scenarios. The urban environments represent anything from a small rural village to a large American-type city centre.

We simulated flights with MATLAB and the UAV toolbox. We used our own implementation of the 3DVFH* in Matlab, as previously discussed in Ref. (40). This version is a baseline with a weighted sum as a cost function. We replaced this cost function with a weighted product, two variants mixing weighted sum and weighted product, a weighted Chebyshev function, and two versions of factorial achievement

https://doi.org/10.1017/aer.2023.68 Published online by Cambridge University Press

scalarising functions with different levels of factorisation ($q = 2$ and $q = 3$ for Equation (6)). The cost of proximity to obstacles plays a special role because it ensures safety. Therefore, we tested the Chebyshev distance and the factorial achievement scalarising function, with $o_{obstacle}$ as part of the cost function and $o_{obstacle}$ as additional cost.

We also investigated the influence of multiple definitions of the objective cost o_i itself. Most optimisation-based path planning algorithms use two or three main types of cost: deviation from the goal direction/a global path, obstacle cost, and smoothness cost. While the first two are often formulated similarly across multiple algorithms, the latter is formulated in multiple ways. Therefore, we investigated the influence of multiple definitions of the smoothness cost o_{smooth} itself. We tested the standard formulation of velocity cost of the 3DVFH* as published in Ref. (39), as well as four alternatives representing different approaches seen in the broader field of local path planning algorithms.

Changing the cost functions also requires the identification of suitable weights. This work also investigates the sensitivity of the solution on the chosen weight. Therefore, we chose a design of experiments (DoE)-inspired approach to get as much information as possible in the shortest time possible. We fully simulate every flight, which means that our tool needs to render the sensor data for every flight position, use the avoidance algorithm for these situations and simulate any physical and dynamic limitations. This approach is computationally heavy. Our limited computational resources do not allow the simulation of millions of flights in a short time. Therefore, we start with a full factorial surface design for every cost function. Sensitive weights are further refined with additional evaluation points in the DoE. The approach quickly identifies statistically significant weights and reduces the parameter room. This trade-off between high-fidelity simulations and optimised experimental design allowed over 600,000 realistic test flights in a reasonable amount of time.

2.7 Performance evaluation

The essential requirement of an obstacle avoidance algorithm is that it has to be safe. Therefore, a collision has to be avoided in any case. An obstacle avoidance algorithm's second most important requirement is to fulfil its mission successfully – its reliability. This work evaluates reliability and safety combined with failure probability. The failure probability f is defined as:

$$f = \frac{n_f}{n_f + n_s} \tag{10}$$

Where n_f denotes the number of failed flights and n_s denotes the number of successful flights. A failed flight is a flight that does not reach its goal. The reason for failure, e.g. crashing or poor path planning, is irrelevant to its reliability. However, we investigate its safety separately.

Besides the algorithm's reliability in fulfilling its mission, its efficiency plays a crucial role, especially in small and nano UAV operations. We use estimated energy consumption as a measure of efficiency. Determining the energy consumption from the simulation is complex and inaccurate. Therefore, we use a simplified approach to only roughly estimate the energy consumption. This approach is simple but sufficient to compare the efficiency of the different cost function types.

The simulation provides accurate position data in x, y and z and velocities in all three directions at 30 Hz. With this information, potential energy and kinetic energy are easy to determine. However, a quadcopter hovering in a fixed position has constant potential energy and no kinetic energy, i.e. no change in energy. In reality, the energy consumption in this situation is the highest. Therefore, a potential and kinetic energy approach is invalid for quadcopter UAVs. A more precise approach based on all motors' power consumption is impossible because the simulation does not obtain this data. However, the required thrust can be estimated from movements. Therefore, the power consumption is estimated with a simple thrust estimation model. This procedure is inspired by the simplified effective hover mode endurance presented in Ref. (41).

https://doi.org/10.1017/aer.2023.68 Published online by Cambridge University Press

The whole flight mission is discretised into time steps. The total energy E_{tot} consumed for a mission is the sum of all energies consumed during the individual time steps E_i.

$$E_{tot} = \sum_i E_i \tag{11}$$

The energy per timestep E_i is the sum of the change in kinetic energy E_{kin}, potential energy E_{pot}, and the energy required to generate a certain thrust E_{thrust}.

$$E_i = \Delta E_{kin,i} + \Delta E_{pot,i} + E_{thrust,i} \tag{12}$$

The velocity and attitude in each time step are assumed to be constant. The energy required for attitude control is neglected. The energy required to accelerate the UAV between two timesteps is estimated by the acceleration work, which is equal to the change in kinetic energy ΔE_{kin}, given by:

$$\Delta E_{kin,i} = \frac{1}{2} \cdot m \cdot \Delta v_i^2 \tag{13}$$

$$\Delta v_i = \sqrt{\left(v_{x,i+1} - v_{x,i}\right)^2 + \left(v_{y,i+1} - v_{y,i}\right)^2 + \left(v_{z,i+1} - v_{z,i}\right)^2} \tag{14}$$

With $\boldsymbol{v} = \left(v_x,\ v_y,\ v_z\right)$ denoting the flight velocity. The energy required for altitude changes is given by the change in potential energy, according to:

$$\Delta E_{pot,i} = m \cdot g \cdot (z_{i+1} - z_i) \tag{15}$$

Finally, the energy required to provide sufficient thrust is derived from the definition of the figure of merit [42] and given by:

$$E_{thrust,\,i} = \frac{T_i^{\frac{3}{2}}}{\sqrt{2 \cdot S_{ref} \cdot \rho}} \cdot \frac{\Delta t}{FOM} \tag{16}$$

Δt is the length of the time step, S_{ref} is the reference area, i.e. the disk area, ρ denotes the air density, and FOM is the figure of merit of the rotors. $FOM = 0.72$ is assumed for the simulation. The thrust of the UAV is denoted by T and estimated via:

$$T_i \cong \sqrt{(m \cdot g)^2 + D^2} \tag{17}$$

m denotes the weight of the UAV, g the gravitational acceleration, and D the drag force acting on the UAV. D is estimated from Ref. (43) with assuming low pitch angles. In all UAV applications, energy is a valuable resource. The lower the energy consumption, the more versatile the UAV application. Energetically very efficient path planning increases the mission range and mission duration. On the contrary, too high energy consumption might limit the UAV in its applicability. Therefore, the influence on energy consumption requires monitoring as well.

The sensitivity of cost function weights is the last parameter to evaluate the different cost function types. If only a specific set of weights leads to good performance while most other parameters lead to bad performance, the cost function has low robustness against wrongly chosen weights. The higher the sensitivity, the lower the robustness of this approach.

2.8 Cost function parameter identification

Local path planning is a two-staged multi-optimisation problem. The local path planning algorithm identifies the best path by considering aspects like the deviation from the goal direction, vicinity to obstacles and path smoothness. However, failure probability, flight distance, flight time and energy consumption measure the algorithm's performance. The cost function parameters are only loosely coupled with the actual performance parameters of the algorithm and do not show deterministic behaviour. Therefore, the ideal choice of cost function parameters is unknown. We conducted a DoE-inspired search for ideal weights. First, we identified a reasonable parameter room for the different cost function weights. Second,

https://doi.org/10.1017/aer.2023.68 Published online by Cambridge University Press

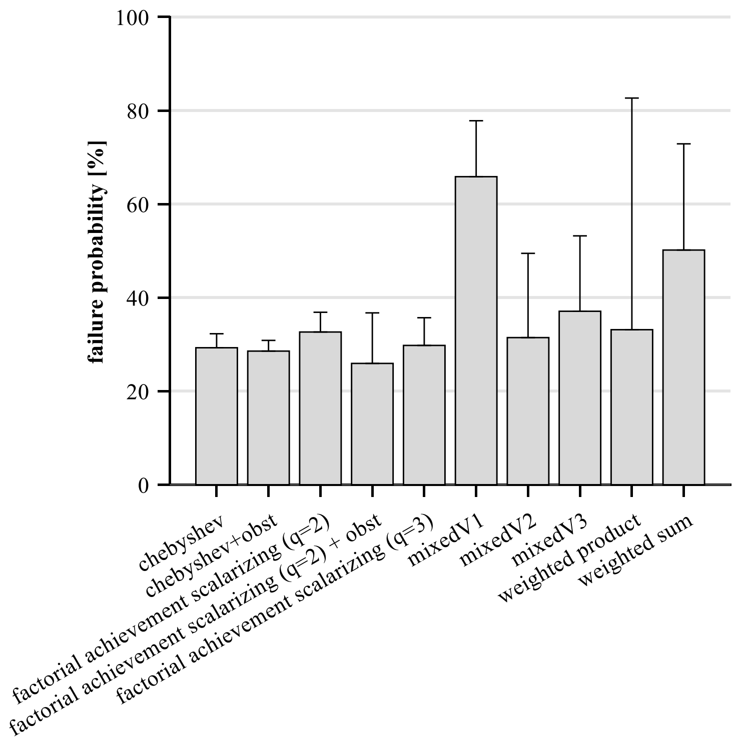

Figure 4. *Comparison of the failure probability of different variants of a cost function for the 3DVFH* for flights in all environments. The bars indicate the minimal failure probability. The error bars indicate the range of failure probabilities.*

we conduct a full factorial surface Design and identify those parameters with a statistically significant influence. Third, those significant parameters are further investigated with a refined sampling around the optimum found by the DoE.

3.0 Results and discussion

This chapter presents and discusses three influences of the tested functions: the influence on safety (failure probability and crash probability), robustness and performance (flight time, distance, and energy consumption). We tested obstacle cost in the weighted Chebyshev and achievement scalarising functions and outside of these functions as additional costs because obstacle costs have a specific role – they evaluate the cost of the vicinity to obstacles.

3.1 Reliability – Failure probability

3.1.1 Factorial achievement scalarising and weighted Chebyshev

Figure 4 shows the failure probability of the weights with the lowest failure probability for each variant. The error bars indicate the failure probability for worse weightings. Using factorial achievement scalarising with a degree of freedom of two as the cost function but considering the obstacle cost as an additional cost has the lowest failure probability (26%) and one of the lowest ranges of failure probabilities (6%). The other variants of the factorial achievement scalarising function and the weighted Chebyshev distance have slightly higher minimal failure probabilities, around 30%. The Chebyshev distance is the most robust function against different choices of objective weighting; however, factorial achievement

https://doi.org/10.1017/aer.2023.68 Published online by Cambridge University Press

scalarising has only slightly higher sensitivities. The weighted Chebysehv distance considers the single worst objective, while the factorial achievement scalarising function with a degree of freedom of two considers the two worst objectives. The factorial achievement scalarising function with a degree of freedom of three considers the three worst objectives. All three function types optimise by minimising bad performance. All three function types have the lowest failure probabilities. All three function types have the lowest sensitivities on the choice of weights. However, Fig. 4 also shows that an increasing number of relevant objectives also leads to increased sensitivity (see Fig. 4 from left two right: one worst objective, one worst objective + obstacle cost, two worst objectives, two worst objectives + obstacle cost, three worst objectives).

3.1.2 Weighted sum and product

Mixing weighted sum and product also yields good performance regarding the minimal failure probability (see Fig. 4 mixedV2). However, it also shows the worst performance depending on the type of mixing (see Fig. 4 mixedV1). In general, all mixed versions of weighted sum and product have a considerably higher sensitivity on the parameters than Chebyshev distance and factorial achievement scalarising. Mixed version V2 has the third lowest failure probability (31.5%). This version sums the costs up, similar to a weighted sum. Exponents perform the weighting, similar to a weighted product. Therefore, objectives with high costs dominate the term. When only the worst objectives dominate the term, it is basically the same as Chebyshev distance or factorial achievement scalarising. However, it depends on the weight, how easily one or multiple objectives dominate the cost term. This dependency explains the higher sensitivity of mixed Version V2 than the Chebyshev distance and factorial achievement scalarising function.

Weighted sum and product try to find the best solution for all objectives, whereas Chebyshev and achievement scalarisation aims at the least bad solution for one or multiple objectives. The analysis shows that the local path planner reaches the goal more often if it focuses on having the least bad objectives rather than the better ones.

3.1.3 Conclusion on the failure probability

Considering and minimising the worst objectives proved the best strategy in our test case. Considering the single worst objective in the Chebyshev distance allows the path planner to automatically consider this objective as the most important, which has the highest impact on the overall result in the current situation. Less important objectives are fully disregarded and do not skew the identification of an ideal solution. Considering the two worst objectives in the factorised achievement scalarising function increases the optimisation's flexibility and gives the objectives' relevance an inherent dynamic to prioritise the right objectives. Consideration of the two worst objectives reaches slightly better failure probabilities than the single worst objective of the weighted Chebyshev distance. A similar output can be generated with a weighted product or a mixture of a weighted product and sum. The disadvantage of the weighted product – that extreme values might overrule the complete function – becomes an advantage in this case. However, weighted product and a mixture of weighted product and sum are more sensitive to weighting than weighted Chebyshev distance and factorial achievement scalarising function.

Looking into specific sub-environments, e.g. only those with few obstacles, those with many high obstacles, etc., shows that the ideal number of relevant worst objectives depends on the environment. Less complex environments, with fewer and smaller obstacles, achieve the lowest failure probability with the weighted Chebyshev distance and factorial achievement scalarising functions with $q = 2$. However, complex environments, e.g. a city centre of a large American city, perform best with $q = 3$. Analysing the flight paths shows that simple environments often have one main problem related to one or two objectives of path planning. By focusing on these objectives (and therefore this single problem) and ignoring other objectives, the problem is quickly solved, and the path is successful. However, in more

https://doi.org/10.1017/aer.2023.68 Published online by Cambridge University Press

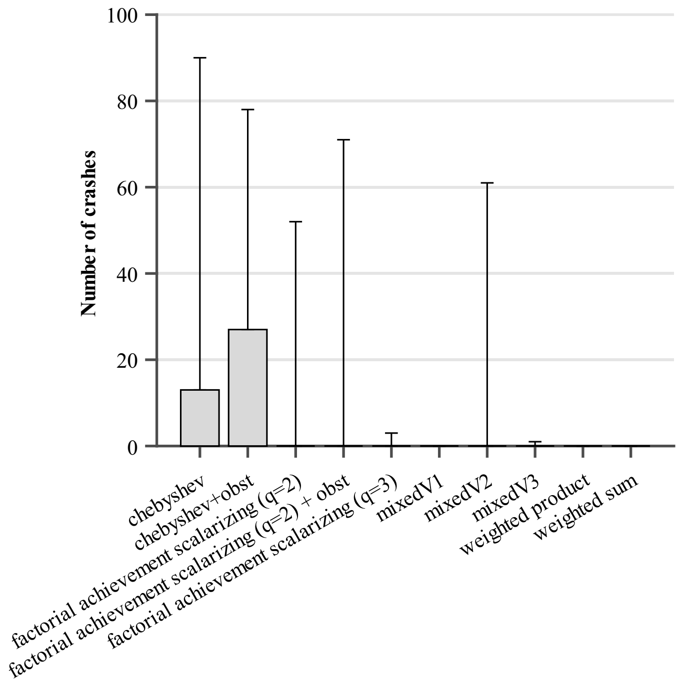

Figure 5. *Comparison of the number of crashes of different variants of a cost function for the 3DVFH* for flights in all environments. The bars indicate the minimal failure probability. The error bars indicate the range of failure probabilities.*

complex environments, a series of problems arise simultaneously. If the focus is on too few or too many objectives, none of the problems is solved, and the path planner gets stuck in an open field.

3.2 Safety – Crash probability

3.2.1 Safety can be measured in two aspects

The previous subchapter already discussed the likelihood of successfully fulfilling a mission. Depending on the mission, a failure might be more or less severe. Mission failure has one of the most severe impacts on medical applications and is particularly important in this work. Nonetheless, the probability of the UAV crashing into an object is also essential for evaluating the different cost function types. Figure 5 shows the number of crashes for the different cost function types and the strongest weakness of the weighted Chebyshev distance – it navigates into buildings. The same problem is seen in the factorial achievement scalarising function and version two and three of mixing weighted product and sum. In these versions, some objectives can easily dominate the whole cost function, leading to similar situations as in the weighted Chebyshev distance. Analysis of the crashed flights shows that the UAV must fly far off the goal direction. A safe distance to obstacles is gradually traded for minimal improvements in the goal direction, which finally leads to a shortfall of a suitable safety distance to obstacles, which we treat as a potential crash. The choice of weights influences how likely the path planner will trade sufficient safety distance for goal-driven planning. The other cost function types, which consider multiple objectives simultaneously, are less prone to crash into buildings. In these algorithms, one or a few objectives cannot easily dominate the cost function. Therefore, other costs do not completely override the cost for the vicinity of obstacles. The two functions with additional obstacle costs (chebshev+obst and factorial achievement scalarising ($q = 2$)+obst in Fig. 5) can be completely dominated by the Chebyshev or factorial achievement scalarising term if the weightings are chosen poorly (upper limit of the error bar).

https://doi.org/10.1017/aer.2023.68 Published online by Cambridge University Press

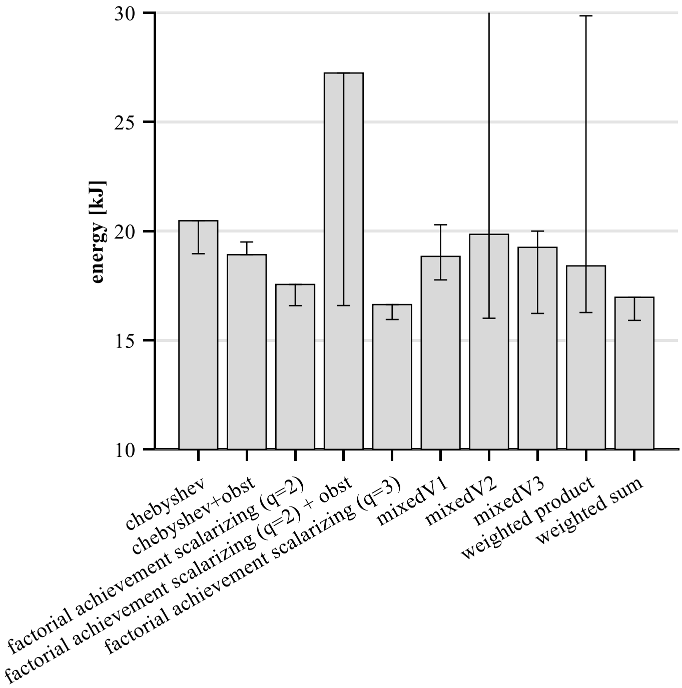

Figure 6. *Comparison of the average energy consumption of different variants of a cost function for the 3DVFH* for flights in all environments. The bars indicate the energy consumption reached with the objective weighting for minimal failure probability/maximum reliability. Error bars indicate the best and worst performance for the individual cost function types with other weightings.*

3.3 Performance – Energy consumption

3.3.1 Factorial achievement scalarising and weighted Chebyshev

Some kind of durability is important for many types of missions. The estimated energy consumption of the performed missions is a good measure of overall efficiency. Figure 6 gives an overview of the estimated average energy consumption for the different cost function types. The weights used for the bars in Fig. 6 are those weights that lead to a minimal failure probability and maximum reliability. Please note that the average energy consumption is based on successful flights only. Therefore, different bars (and the error bars) are based on partly different environments. The different environments skew the analysis to a certain degree.

Figure 6 shows that the average energy consumption decreases with an increasing number of relevant objectives, except for factorial achievement scalarising ($q = 2$) + obstacle cost. However, the error bar also indicates that function type can have a lower average energy consumption. That consideration of more criteria leads to more efficient flight paths is not surprising. One criterium ensures safety (obstacle cost). All other criteria improve efficiency in one way or another – either by ensuring short paths or minimal deviation from the goal direction or by ensuring smooth paths requiring fewer control inputs. Its low failure probability explains the high average energy consumption of the factorial achievement scalarising ($q = 2$) and additional obstacle cost function. This function finds suitable paths in environments in which all other approaches fail. The energy consumption of some of these paths is up to ten times higher than the average of other paths. These paths are certainly not efficient. Nevertheless, they are at least valid. These paths lead to the relatively high average energy consumption of this function. Looking only at environments where all three versions of the factorial achievement scalarising function succeed has around the same average energy consumption as $q = 2$ without additional obstacle cost. The error bar of factorial achievement scalarising ($q = 2$) + obst also indicates that this function can have the

https://doi.org/10.1017/aer.2023.68 Published online by Cambridge University Press

same average energy consumption as the other factorial achievement scalarising functions. However, in these cases, the failure probability is slightly higher.

3.3.2 Weighted sum and product

Weighted product and weighted sum vary greatly in efficiency, depending on the chosen weights. The UAV has two options to evade an obstacle: fly around or over the obstacle. By setting specific weights for yaw and pitch cost, the operator chooses which of the two avoidance manoeuvres is preferred to a certain degree; however, the environment defines which manoeuvre is best. Sometimes, the algorithm is very efficient in a specific environment but fails in all other environments. In these cases, the average energy consumption is quite low (low lower error bar). Therefore, most functions' lowest average energy consumption is roughly the same. Poor weights can also lead to very high energy consumption at simultaneously lower failure probability than possible for this function type (upper error bars of the five right bars in Fig. 6).

3.3.3 Conclusion on the energy consumption

As long as energy is one of the most limiting factors of small UAV, its consumption always needs to be considered. While low energy consumption can increase a system's versatility significantly, high energy consumption can render a system completely unusable. Especially in medical applications, low energy consumption also allows the transportation of additional equipment – increasing the benefits of the UAV further. Figure 6 shows that focusing on a few objectives (e.g. with the weighted Chebyshev distance) leads to significantly higher energy consumption compared to other approaches. Even though this approach has a relatively low failure probability, its high energy consumption might render it unusable for some applications. However, Fig. 6 also shows that a well-chosen factorial achievement scalarising function, for example, one with a degree of freedom of three, is better than the current standard, the weighted sum.

3.4 Performance – Flight distance

3.4.1 Factorial achievement scalarising and weighted Chebyshev

Figure 7 shows the minimal path length for the different cost function types. Only successful flights are considered. Similar to the energy consumption and what is seen in Fig. 6, an increasing number of relevant parameters decrease the average distance. Again, factorial achievement scalarising ($q = 2$) with obstacle cost is an exception. The UAV flies long detours to find paths in worlds where other functions get stuck; however, a slight adaption of objective weighting leads to a strong reduction of flight distance (lower error bar) at the cost of slightly increased failure probability.

3.4.2 Weighted sum and product

Weighted sum, product and mixtures of those vary less in-flight distance than in energy consumption. Again, efficiency depends on the parameter setting, and the operator can easily tune the parameters such that the UAV finds short paths in a certain environment. However, the algorithm is too specific for an environment such that it fails in many other environments. A wrong choice of parameters can also lead to long detours and inefficient path planning.

3.4.3 Conclusion on the flight distance

Long flight paths are less efficient and potentially disturb more people on the way. Figure 7 shows that most functions lie relatively close together. The type of cost function only has a small effect on the path length.

https://doi.org/10.1017/aer.2023.68 Published online by Cambridge University Press

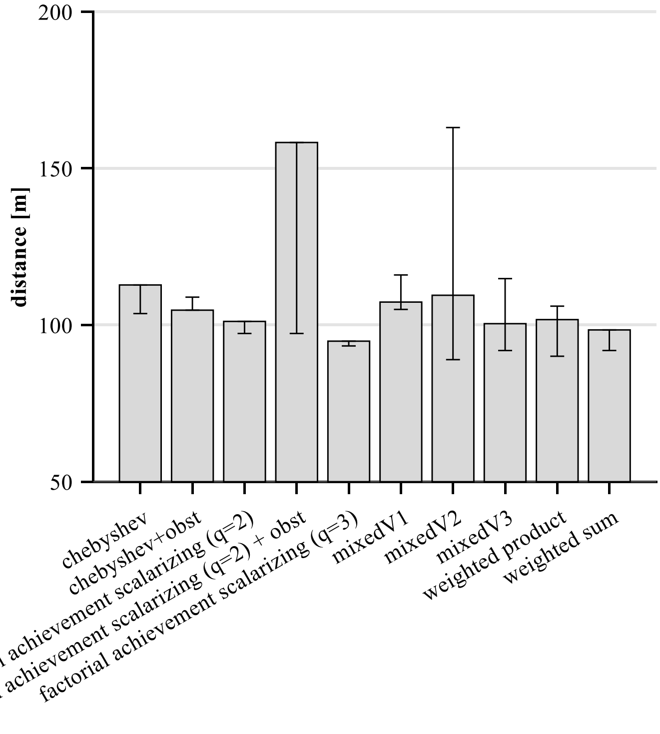

Figure 7. *Comparison of the average flight distance of different variants of a cost function for the 3DVFH* for flights in all environments. The bars indicate the flight distance reached with the objective weighting for minimal failure probability/maximum reliability. Error bars indicate the best and worst performance for the individual cost function types with other weightings.*

3.5 Performance – Flight time

3.5.1 Factorial achievement scalarising and weighted Chebyshev

Figure 8 shows the flight time for the different cost function types; however, only successful flights give a meaningful flight time. Additionally, Fig. 8 shows the lowest reachable flight time for every cost function type, independent of the failure probability, indicated by the error bars.

Like the other performance charts in Figs 6 and 7, factorial achievement scalarising with a degree of freedom of 3 has the lowest flight time. The more objectives are simultaneously relevant, the better the result for those functions minimising bad performance. Again, factorial achievement scalarising ($q = 2$) with obstacle cost is an exception because of the more complicated missions it solves. In general, the variability in flight time is also relatively small for the Chebyshev distance and factorial achievement scalarising functions.

3.5.2 Weighted sum and product

Weighted sum, product and mixtures of both show a wide range of average flight times; notably, weighted product and mixedV2 are very sensitive to parameter tuning. Low average flight times are achieved with ideal weights for specific environments. Other environments fail and are irrelevant for Fig. 8, leading to low lower error bars. Weighted product and mixed V2 also have very high upper error bars. Therefore, at least one set of weights must be inefficient in most environments. These two function types are most likely to overvalue extremes. They are also easily dominated by a specific objective if the weight is poorly chosen. For example, very high obstacle cost weight makes the path planner extremely careful and leads to long flight times.

https://doi.org/10.1017/aer.2023.68 Published online by Cambridge University Press

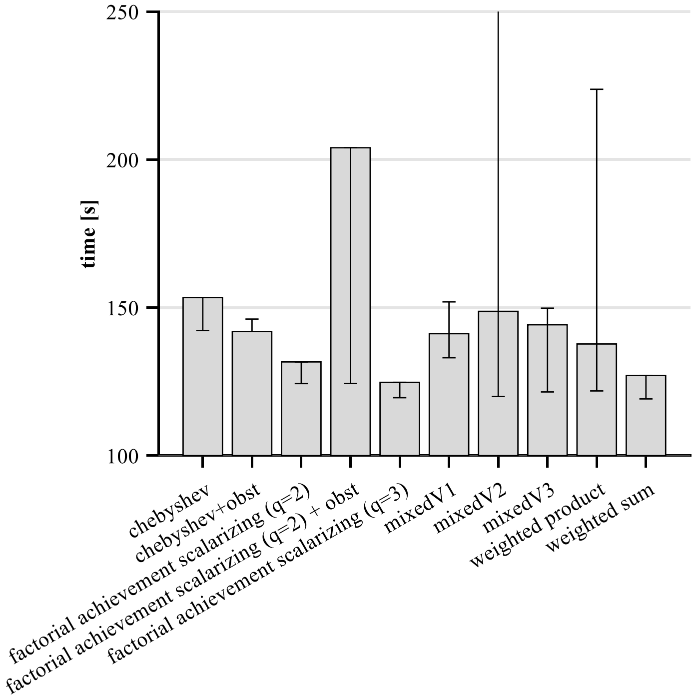

Figure 8. *Comparison of the average flight time of different variants of a cost function for the 3DVFH* for flights in all environments. The bars indicate the flight time reached with the objective weighting for minimal failure probability/maximum reliability. Error bars indicate the best and worst performance for the individual cost function types with other weightings.*

3.5.3 Conclusion on the flight time

The flight time is particularly important for medical applications. Here, the different cost function types show clear differences. The weighted sum is relatively efficient in-flight time. However, a factorial achievement scalarising function with a degree of freedom of three has a similar flight time but considerably lower failure probability and energy consumption. Therefore, using a weighted sum yields no advantages over a factorial achievement scalarising function with a degree of freedom of three.

4.0 Conclusion and outlook

Formulating a robust, reliable and energy-efficient path planner for collision avoidance manoeuvers is a key aspect in developing increasingly autonomous unmanned air vehicles (UAV). The need to simultaneously address several physical laws and conflicting optimisation goals leads to multi-objective and multi-constrained optimisation problems, which, coupled with the significant nonlinearities of flight dynamics, environmental (e.g. weather) and other models, ultimately results in non-convex and numerically intricate solution spaces. This work demonstrated the potential of various cost functions in multi-objective optimisation problems for local path planning in various environments. We showed that functions that aim to minimise the worst performance at every step have a lower failure probability than those that try to maximise good performance in objective functions. A factorial achievement scalarising function with a degree of freedom of two and additional cost for vicinity to obstacles has the lowest failure probability. Other forms of factorial achievement scalarising functions and the weighted Chebyshev distance have slightly higher failure probabilities. While a mixture of weighted product and sum can reach similarly low failure probabilities, the mixed cost function is sensitive to the correct choice of

https://doi.org/10.1017/aer.2023.68 Published online by Cambridge University Press

objective weights. The weighted Chebyshev distance and the factorial achievement scalarising function are very robust and not as sensitive to the choice of weights as the other methods. Compared to the factorial achievement scalarising function, the weighted Chebyshev distance shows slight disadvantages regarding flight time, distance and energy consumption. Taking additional objectives into account and minimising those with the highest costs leads to better overall performance.

However, a verification based on the 3DVFH* and the investigated cost functions alone is insufficient for the generality of path planning application cases and does not capitalise on the advantages offered by machine learning implementations. Therefore, further work is recommended to improve UAV collision avoidance path planning algorithms. In particular, combining the 3DVFH* with a better cost function and additional improvements, e.g. bio-inspired flight strategies or a neural network, might decrease the failure probability further. Investigating the influence of the cost function type on other algorithms with a better base performance than the 3DVFH*, e.g. FASTER, is also a potential future research topic.

Competing interests. The authors declare no competing interests.

Funding. No funding was received for this specific project.

References

[1] Marshall, D.M. *UAS Integration into Civil Airspace: Policy, Regulations and Strategy*. Hoboken, New Jersey: John Wiley & Sons, Inc., 2022.
[2] Politi, E., Panagiotopoulos, I.E., Varlamis, I. and Dimitrakopoulos, G.J. A survey of UAS technologies to enable Beyond Visual Line Of Sight (BVLOS) operations, In: *International Conference on Vehicle Technology and Intelligent Transport Systems*, 2021.
[3] Ramesh, P.S., Jeyan, J.V. and Lal, M. Comparative analysis of the impact of operating parameters on military and civil applications of mini unmanned aerial vehicle (UAV), In: *AIP Conference Proceedings,* 2020.
[4] Basiri, A., Mariani, V., Silano, G., Aatif, M., Iannelli, L. and Glielmo, L. A survey on the application of path-planning algorithms for multi-rotor UAVs in precision agriculture, J. Navigat., 2022, 75, (2), pp 364–383.
[5] Aggarwal, S. and Kumar, N. Path planning techniques for unmanned aerial vehicles: A review, solutions, and challenges. In: Computer communications, Jg. 149, 2020, S. 270–299.
[6] Pittner, M., Hiller, M., Particke, F., Patino-Studencki, L. and Thielecke, J. Systematic analysis of global and local planners for optimal trajectory planning, In: *ISR* 2018*; 50th International Symposium on Robotics*, 2018.
[7] Patle, B.K., Ganesh Babu, L., Pandey, A., Parhi, D.R.K. and Jagadeesh, A. A review: On path planning strategies for navigation of mobile robot, *Defence Technol.*, 2019, **15**, (4), pp 582–606. https://www.sciencedirect.com/science/article/pii/S2214914718305130.
[8] Khan, S.I., Qadir, Z., Munawar, H.S., Nayak, S.R., Budati, A.K., Verma, K.D. and Prakash, D. UAVs path planning architecture for effective medical emergency response in future networks, *Phys. Commun.*, 2021, 47, p 101337. https://www.sciencedirect.com/science/article/pii/S1874490721000744
[9] Marler, R.T. and Arora, J.S. Survey of multi-objective optimization methods for engineering, *Struct. Multidiscip. Optim.*, 2004, **26**, (6), pp 369–395.
[10] Gardi, A., Sabatini, R. and Ramasamy, S. Multi-objective optimisation of aircraft flight trajectories in the ATM and avionics context, *Prog. Aerosp. Sci.*, 2016, **83**, pp 1–36. https://www.sciencedirect.com/science/article/pii/S0376042115300105
[11] Gardi, A., Sabatini, R., Ramasamy, S. and Kistan, T. Real-time UAS guidance for continuous curved GNSS approaches, *J. Intell. Robot. Syst.*, 2019, **93**, (1), pp 151–162.
[12] Qin, H., Shao, S., Wang, T., Yu, X., Jiang, Y. and Cao, Z. Review of autonomous path planning algorithms for mobile robots, *Drones*, 2023, **7**, (3). https://www.mdpi.com/2504-446X/7/3/211
[13] Aggarwal, S. and Kumar, N. Path planning techniques for unmanned aerial vehicles: A review, solutions, and challenges, *Comput. Commun.*, 2020, **149**, pp. 270–299.
[14] Tordesillas, J., Lopez, B.T., Everett, M. and How, J.P. FASTER: Fast and Safe Trajectory Planner for Navigation in Unknown Environments, *IEEE Trans. Robot.*, 2022, **38**, (2), pp. 922–938.
[15] Sang, H., You, Y., Sun, X., Zhou, Y. and Liu, F. The hybrid path planning algorithm based on improved A and artificial potential field for unmanned surface vehicle formations. *Ocean Eng.*, 2021, **223**, p 108709.
[16] Tordesillas, J. and How, J.P. PANTHER: Perception-aware trajectory planner in dynamic environments, *IEEE Access*, 2022, **10**, pp 22662–22677.
[17] Li, Z., Arslan, Ö. and Atanasov, N. Fast and safe path-following control using a state-dependent directional metric, In: *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pp 6176–6182, 2020.
[18] Golroudbari, A.A. and Sabour, M.H. *Recent Advancements in Deep Learning Applications and Methods for Autonomous Navigation: A Comprehensive Review*, 2023.
[19] He, L., Aouf, N. and Song, B. Explainable deep reinforcement learning for UAV autonomous path planning, *Aerosp. Sci. Technol.*, 2020, **118**, p 107052.

https://doi.org/10.1017/aer.2023.68 Published online by Cambridge University Press

[20] Shin, S.-Y., Kang, Y.-W. and Kim, Y.-G. Obstacle avoidance drone by deep reinforcement learning and its racing with human pilot, *Appl. Sci.*, 2019, **9**, (24). https://www.mdpi.com/2076-3417/9/24/5571
[21] Padhy, R.P., Verma, S., Ahmad, S., Choudhury, S.K. and Sa, P.K. Deep neural network for autonomous UAV navigation in indoor corridor environments, *Proc. Comput. Sci.*, 2018, **133**, pp 643–650. https://www.sciencedirect.com/science/article/pii/S1877050918310524
[22] Sabatini, R., Roy, A., Blasch, E., Kramer, K.A., Fasano, G., Majid, I., Crespillo, O.G., Brown, D.A. and Ogan Major, R. Avionics systems panel research and innovation perspectives, *IEEE Aerosp. Electron. Syst. Mag.*, 2020, **35**, (12), pp 58–72.
[23] Lizzio, F.F., Capello, E. and Guglieri, G., A review of consensus-based multi-agent UAV applications, In: *2021 International Conference on Unmanned Aircraft Systems (ICUAS)*, pp 1548–1557, 2021.
[24] Puente-Castro, A., Rivero, D., Pazos, A. and Fernandez-Blanco, E. A review of artificial intelligence applied to path planning in UAV swarms, *Neural Comput. Appl.*, 2022, **34**, (1), pp 153–170.
[25] Khoufi, I., Laouiti, A. and Adjih, C. A survey of recent extended variants of the traveling salesman and vehicle routing problems for unmanned aerial vehicles. *Drones*, 2019, **3**, (3). https://www.mdpi.com/2504-446X/3/3/66
[26] Liu, H., Ge, J., Wang, Y., Li, J., Ding, K., Zhang, Z., Guo, Z., Li, W. and Lan, J. Multi-UAV optimal mission assignment and path planning for disaster rescue using adaptive genetic algorithm and improved artificial bee colony method, *Actuators*, 2022, 11, (1). https://www.mdpi.com/2076-0825/11/1/4
[27] Jain, G., Yadav, G., Prakash, D., Shukla, A. and Tiwari, R. MVO-based path planning scheme with coordination of UAVs in 3-D environment, *J. Comput. Sci.*, 2019, **37**, p 101016. https://www.sciencedirect.com/science/article/pii/S1877750318310263
[28] Chen, Y., Luo, G., Mei, Y., Yu, J. and Su, X. UAV path planning using artificial potential field method updated by optimal control theory, *Int. J. Syst. Sci.*, 2016, **47**, (6), pp. 1407–1420.
[29] Seo, D. and Kang, J. Collision-avoided tracking control of UAV using velocity-adaptive 3D local path planning, *Int. J. Cont. Automat. Syst.*, 2023, **21**, (1), pp 231–243.
[30] Ramasamy, S., Sabatini, R., Gardi, A. and Liu, J. LIDAR obstacle warning and avoidance system for unmanned aerial vehicle sense-and-avoid, *Aerosp. Sci. Technol.*, 2016, **55**, pp 344–358. https://www.sciencedirect.com/science/article/pii/S1270963816301900
[31] Branke, J., Deb, K., Miettinen, K. and Słowiński, R. *Multi-objective Optimization: Interactive and Evolutionary Approaches*, 1st edition. Berlin, Germany: Springer, 2008.
[32] Marler, R.T. and Arora, J.S. The weighted sum method for multi-objective optimization: New insights. *Struct. Multidiscipl. Optim.*, 2010, **41**, (6), pp 853–862.
[33] San Cristóbal Mateo, J.R. Weighted sum method and weighted product method, In: *Multi Criteria Analysis in the Renewable Energy Industry*. London: Springer London, pp 19–22, 2012.
[34] Tofallis, C. Add or multiply? A tutorial on ranking and choosing with multiple criteria, *Trans. Educ.*, 2014, **14**, (3), pp 109–119.
[35] Cantrell, C.D. Modern Mathematical Methods for Physicists and Engineers. Cambridge, UK: Cambridge University Press, 2000.
[36] Wierzbicki, A.P. The use of reference objectives in multi-objective optimization, In: *Multiple Criteria Decision Making Theory and Application*. Berlin, Heidelberg: Springer Berlin Heidelberg, pp 468–486, 1980.
[37] Nikulin, Y., Miettinen, K. and Mäkelä, M.M. A new achievement scalarizing function based on parameterization in multi-objective optimization. *OR Spectr.*, 2012, **34**, (1), pp 69–87.
[38] Mavrotas, G. Effective implementation of the ε-constraint method in Multi-objective mathematical programming problems, *Appl. Math. Comput.*, 2009, **213**, (2), pp 455–465.
[39] Baumann, T. Obstacle avoidance for drones using a 3DVFH algorithm, Masters Thesis, 2018.
[40] Thomeßen, K., Thoma, A. and Braun, C. Bio-inspired altitude changing extension to the 3DVFH* local obstacle avoidance algorithm, In: *Deutsche Gesellschaft für Luft- und Raumfahrttechnik (Hrsg.)Deutscher Luft- und Raumfahrtkongress*, 2022.
[41] Dorrington, G.E. Performance of Electric Vertical Take-Off and Landing (EVTOL) hovering craft, In: *Engineers Australia (Hrsg.)18th Australian International Aerospace Congress, Melbourne,* 2019. https://search.informit.org/doi/10.3316/informit.319952454291139
[42] Leishman, J.G. Principles of Helicopter Aerodynamics, 2nd edition. New York, NY: Cambridge University Press, 2005.
[43] Russell, C., Jung, J., Willink, G. and Glasner, B. Wind tunnel and hover performance test results for multicopter UAS vehicles, In: *Internationa Annual Forum and Technology Display*, 2016.

Cite this article: Thoma A., Thomessen K., Gardi A., Fisher A. and Braun C. (2023). Prioritising paths: An improved cost function for local path planning for UAV in medical applications. *The Aeronautical Journal*, **127**, 2125–2142. https://doi.org/10.1017/aer.2023.68