Multi-objective modular strategic planning framework for low altitude missions within the Urban Air Mobility ecosystem

FLAVIA CAUSA, Member, IEEE

GIANCARMINE FASANO, Senior Member, IEEE

University of Naples "Federico II", Naples, Italy

***Abstract*—This paper introduces a multi-metric multi-constraint strategic path planning framework applicable to unstructured urban airspace. The planner is based on a modular and scalable approach to handle several information sources and aspects characterizing urban flight scenarios, such as risk and weather maps, landing site locations, navigation requirements, and mobile and fixed obstacle characteristics. This information is coupled with dynamic constraints and UAV specifications to derive a flyable and safe path connecting a start position and a destination. Strategies for data gathering and synthesis, used to keep a reduced computational burden, are described along with the path planner algorithm. The latter consists in three steps specifically developed to handle both static and time-varying information. A multi-objective cost function with variable weighting coefficients has been implemented so that the most relevant factors for the considered applications can be selected in an adaptive fashion. The performance of the developed algorithms is tested by investigating the planner behavior when changing its inputs as well as the cost function weighting coefficients, demonstrating the ability of the planner in returning an efficient and safe trajectory.**

Manuscript received XXXXX 00, 0000; revised XXXXX 00, 0000; accepted XXXXX 00, 0000.

This work has been carried out in the framework of the project SMARTGO, funded by the Italian Space Agency (ASI).

Authors' address: The authors are with the Department of Industrial Engineering, University of Naples Federico II, 80125 Naples, Italy, E-mail: (flavia.causa@unina.it; giancarmine.fasano@unina.it). (Corresponding author: Flavia Causa).

Color versions of one or more of the figures in this article are available online at http://ieeexplore.ieee.org.

I. INTRODUCTION

Urban Air Mobility (UAM) represents an opportunity-rich scenario for Unmanned Aircraft Systems (UAS), which are expected to carry out several missions in urban environments in highly autonomous fashion, in the next future. The growing requests for operations to be carried out by Unmanned Aerial Vehicles (UAVs) in very low altitude and people dense environments calls for improved planning and decision-making capabilities to ensure safe integration of these platforms. In addition, as the number and density of UAVs increase, more sophisticated airspace management [1] and traffic control strategies are needed. Trajectory design in these environments is not a trivial task. It consists in defining the trajectory connecting a start and a goal point which can be flown by the vehicle. Fixed and mobile obstacles, navigation performance, and dynamic constraints of the UAV must be taken into account during trajectory design to ensure path flyability.

In order to optimize the path while also reducing the computational burden, path definition usually includes a strategic (i.e., offline) phase and a tactical (i.e., online) replanning step which modifies the strategic trajectory in case an unpredicted event (e.g., incoming conflict with an *a priori* unknown obstacle) can negatively impact its safety. An example of this strategy is reported in [2], where strategic path definition carried out with RRT* algorithm is followed by a tactical deconfliction strategy which uses precomputed trajectories to avoid collisions with mobile obstacles. Strategic phase uses all the a-priori information to plan an optimal path, and its output is strictly related to the quality and the quantity of available information, thus making information gathering a crucial step of the planning phase. The more information is available at strategic level, the more it is unlikely that the path has to be modified during the tactical phase. Several services are now under development for retrieving and predicting information related to the urban environment, such as GNSS coverage and integrity [3]–[5], ground risk [6]–[8], and weather [9] and/or wind [10] information. But other information regarding urban environments and strictly connected to the airspace should be considered, such as airspace structure [11], [12] and landing site locations.

Airspace structure highly influences the path planning solution, due to the imposed rules over operating areas. Several airspace concepts have been defined so far ranging from unstructured (allowing free flight) to structured configuration where the usage of geovectoring, zones or lanes and tube approaches have been considered [11].

Landing site definition is a widely investigated problem in the open literature, where several strategies have been developed for correctly tailoring landing site selection to environment topography and ground risk [13], [14]. Besides optimally selecting ground landing site, other approaches have been considered which only assume as landing opportunities vertiport locations [15].

Risk and weather data are generally used within the path construction phase either as information to drive the trajectory optimization, or to define areas the UAV cannot cross, e.g., regions of space where the wind speed is higher than the maximum admissible by the vehicle, or the risk is too high. The problem of defining the trajectory which minimizes the ground risk has been widely investigated in the open literature both in 3d [16] and 2d scenario [17]. Risk information is usually derived using population data over different weekdays, obtained by census data [18], real time sensing [19], phone activity. However also models involving multi-information risk assessment, thus combining population information with urban environment topology have been used in [7], [8]. Weather-based path definition has been tackled both in strategical and tactical phase in [20]. Here, wind module information has been used either to optimize the overall flight time [21], [22] or to minimize the energy consumption of the UAV [22]–[24].

As far as the navigation performance is concerned, the obstacle dense urban environment requires a small UAV localization error while negatively impacting, at the same time, the performance of classic GNSS/INS fusion strategies. Indeed, GNSS signals are deviated and or obstructed by the 3D environment, potentially producing unreliable and large errors in navigation solution. To meet required navigation performance (RNP) levels, several solutions have been developed in the open literature which either exploit a selection mechanism of GNSS signals [25], [26] or use additional information coming from exteroceptive sensors, such as LiDAR [27], cameras [28], or cooperative platforms [29], [30]. Besides these approaches, a very common solution when dealing with path planning consists in building a navigation-aware trajectory, which aims at fulfilling RNP and minimizing the navigation error covariance. Following this approach several works developed path planning solutions based on Bellman-Ford [31], A* [32] or Partially Observable Markow Decision Process (POMDP) [33].

To overcome the computational complexity of the aforementioned graph-based algorithms, which increases with the environment size, the authors developed in [5] a navigation-aware planning strategy using a sampling based approach, i.e. Batch Informed Tree (BIT*), and GNSS coverage maps. The navigation-aware trajectory was chosen among several alternatives (derived as a function of the GNSS coverage conditions), as the shortest one fulfilling the RNP.

Following this line of research, this paper aims at extending the work carried out in [5] to develop a multi-metric multi-constraint strategic planning framework able to deal with several constraints and information inputs which are relevant to the urban environment. This framework is specifically designed for rotorcraft and has been developed within the project SMARTGO (gnsS-enabled urban air Mobility through Ai-powered environment-awaRe Techniques for strateGic and tactical path planning Operations) funded by the Italian Space Agency and coordinated by the University of Naples.

The paper provides the following contributions:

- It develops a modular strategic planning framework which can handle both mobile and fixed obstacles, and can provide a trajectory which fulfills the airspace constraints while being navigation, risk, and weather aware;
- Due to the significant amount of information the planner has to deal with, an innovative data simplification approach is introduced, so that some of the planner inputs can be stored in maps.
- Path geometry can be shaped by a weighted multi-optimization cost function which can be adapted to the user needs;
- Path planning strategy is tested in a wide city-like environment based on a real world scenario. Algorithm performance is assessed by changing input information and cost weight.

The strategic algorithm detailed in this paper is part of a broader planning pipeline developed within the SMARTGO project, composed by a strategic (off-line) path definition and a tactical path modification if an unpredicted event should compromise the path safety during the flight. Details about the entire planning framework are reported in [34].

To the best of the authors' knowledge, this paper represents the first attempt of combining all the considered information sources and constraints for path generation in UAM relevant environments. A multi-optimization multi-constrained algorithm using GNSS/Lidar, and risk maps have been proposed in [18], while using constant altitude routes. Following a similar line of reasoning, the current paper enables full 3d planning and adds several information inputs including static risk maps, landing site maps, navigation performance (depending on GNSS coverage), wind maps and dynamic traffic information which represent a wide range of information sources and constraints relevant for path generation in UAM relevant environments.

The remaining of the paper is structured as follows. Section II introduces the modular inputs that the planner

has to take into account and describes data simplification approaches. Hence, the developed planning framework is detailed in section III. Section VI reports results obtained in simulations considering a real-world scenario. Finally, section V draws the conclusion of this work.

II. CONSTRAINTS AND PLANNING INPUTS

As mentioned earlier, the requirements and constraints to be accounted for when defining a safe path to be flown in urban environment are multiple and of different nature. Following a scalable approach, the conceived planner has to be able to work with the whole set of information, or with a subset of it. Environmental, airspace and vehicle constraints must be taken into account to ensure the planned path meets safety and flyability requirements.

Environmental constraints are mostly related to the characteristics of the scenario and include 3D obstacles, ground risk, and weather/wind forecast.

Airspace constraints are all the constraints which are related to the traffic and airspace management system. Dealing with the traffic is the most complex aspect. Indeed, the path planner should ensure a conflict-free trajectory, even with increasing traffic density. On the other hand, airspace management system constraints possibly include areas for landing and contingency landing (e.g., vertiports) which limit the departure and arrival locations of the UAVs, as well as airspace rules, no-fly zones, and altitude limitations.

Path flyability is also linked with **vehicle constraints**. They include dynamic constraints (i.e., maximum velocity, limits on flight path angle, path length, and endurance) and autonomy constraints. The latter ones are specifically accounted for to ensure that the UAV can fly the mission in an autonomous way, and are strictly related to navigation and control errors in following the trajectory. Navigation and control error bounds must be taken into account when defining separation minima of the UAV with respect to moving and fixed obstacles. In addition, navigation error propagation along the trajectory is needed to verify that the navigation error is always consistent with the RNP.

A. Constraints simplification

Since most of the optimization techniques have been developed to deal with spatial based representations, 3D and 2D maps are used to enclose and simplify some of the aforementioned constraints. These include:

1) *3D Maps of the fixed obstacles and No-fly zones (NFZ)* obtained from publicly available sources and the associated zone classification information which can be retrieved from CityGML database [35].
2) *2D Risk maps* which associate to each ground coordinate a risk level. Although the planning framework can be adaptive to any risk map information, in this work they are derived by classifying the terrain using satellite imagery, and combining this information with the knowledge of the building position and the CityGML-based information, as detailed in [34]. Six levels of risk are envisaged, in ascending order. Level 1 and 2 include natural and rural areas, and industrial areas, respectively, which are characterized by low population density. Level 3 and 4 are associated with cities, with a lower level of risk assumed over the building roofs (i.e., level 3) due to sheltering factor considerations. Level 5 includes high risk areas where UAV flight should be avoided if possible (e.g., hospitals, power plants and railway stations). Finally, areas where flight is forbidden (airports and military areas) is included in Level 6, which can be included within the No fly zones. Especially considering levels 3 and 4, it is worth remarking that the level of risk in urban areas may vary as a function of the forecasted population density as done in [8].
3) *2D Maps containing information about distance from the landing sites.* They are cost maps derived from the knowledge of the landing sites, whose location is known and fixed. A lower cost is associated to the points whose horizontal distance from the landing sites is smaller.
4) *Multi-dimensional wind maps.* They associate the 3D information about forecasted wind module and direction to each ground coordinate of the environment. This information impacts energy costs and may also define no fly zones when the conditions are beyond the worst ones that are acceptable for safe flight. Due to the granularity in space and time of the current available wind maps, a static map is considered in this paper which uses only 2D variation of the wind information, assuming it constant in the vertical direction.
5) *GNSS Coverage Maps,* which are 2.5D elevation maps defining for each point of the environment the altitude at which the position dilution of precision (here referred to simply as DOP) becomes lower than a threshold. A discrete number (J) of GNSS coverage maps is defined corresponding to various DOP levels. In general, these elevation maps inflate for decreasing DOP value. They are derived as detailed in [5], and retained constant along the whole mission. GNSS coverage maps synthesize the navigation performance of a GNSS receiver in a simplified and concise information, avoiding the need to propagate the navigation error when building the trajectory.
6) *Traffic information* known at strategic level, which includes the 4D trajectory of the mobile obstacles. This information is lumped in time-varying 3D maps. The latter, being of the same nature of the fixed obstacles

map, allow collapsing them in a single information when performing collision checks. Being T_{tot} the total endurance of the vehicle whose flight plan has to be built, information about other vehicles' trajectories is stored in N_o occupancy maps (OM) each one covering an interval of time of duration Δt.

$$\Delta t = T_{tot} / N_o \tag{1}$$

Specifically the n_o-th (n_o = 1,…,N_o) map is used by the planner as mobile obstacle occupancy map in the interval of time going from t_{no-1} to t_{no}, assuming that $t_{no} = t_0 + n_o\Delta t$ and t_0 is the start time of the mission. This occupancy map includes the volumes which are assumed to be reserved to the mobile obstacles (called "intruders") and cannot be crossed by any other UAV. These volumes are obtained by offsetting with a margin both in position and time the intruder trajectories in the n_o-th interval as detailed in [36]. The position offset is obtained by assuming the intruder is enclosed in each point of the trajectory in a sphere with radius d_c. The time offset, defined as Δt_c, is used to account for intruder timing uncertainties and to increase the safety distance along the intruder direction of motion.

Some of the information stored within the maps can either vary as a function of the initial time of the mission, i.e., t_0, such as the GNSS coverage maps [5] and the traffic information, or can be dependent on the safety level required by the user. As an example, level 5 areas can be included in the NFZ if a more stringent requirement on risk is demanded by the user. An example of the maps described above is shown in Fig. 1. Fig. 1a and b show the lateral and top view of fixed obstacles (gray) and NFZ (orange) maps. In order to contextualize the data, 3D obstacles are also reported in all the other subfigures. Normalized landing site maps and risk maps are reported in Fig. 1c and d, respectively. Risk maps are characterized by two colors in this case, highlighting in black the buildings which are associated with the lowest cost. Landing sites map's intensity increases with the distance from the landing sites (depicted as red asterisks). The wind map reports the module and the direction of the wind, a top view representation is shown in Fig. 1e. Fig. 1f reports the time varying occupancy map of a single mobile obstacle. The blue color delineates the volume which encloses the trajectory of the intruder (which is the union of all the time varying occupancy maps). The occupancy volume which is accounted for in the 3rd time-varying map is highlighted in red. The trajectory of the intruder during the time interval t_2-t_3 is reported as a grey line to highlight the positioning and time offset used to generate the time-varying maps. Finally, the 3D GNSS coverage map corresponding to a DOP value of 4 is reported in Fig. 1g.

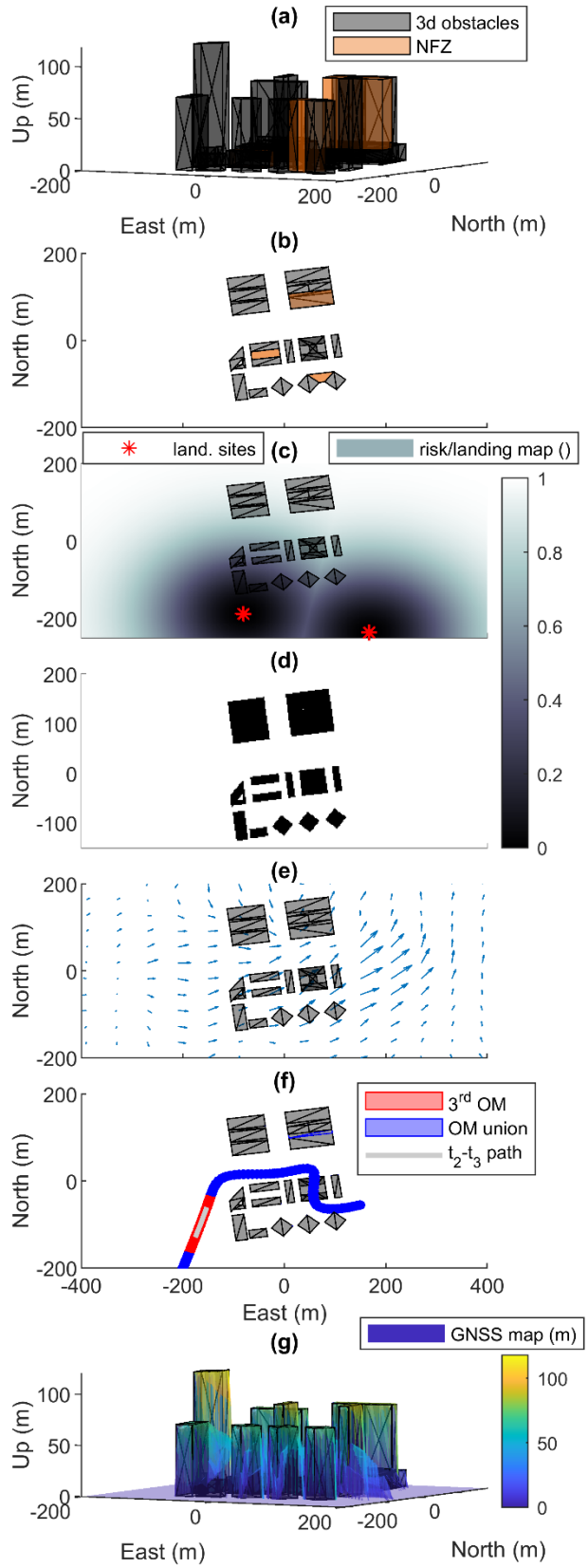


Fig. 1. Example of planning inputs. (a) 3D and (b) topview representation of fixed obstacles and NFZ. (c) Landing site (d) risk and (e) wind maps top views. (f) Mobile obstacles maps: volume enclosing the obstacle trajectory (blue) and 3rd time-varying occupancy map (red). (g) 3d representation of GNSS coverage map obtained assuming DOP = 4.

III. STRATEGIC PATH PLANNING ALGORITHM

The strategic path planning algorithm, whose flowchart is reported in Fig. 2, consists of three steps specifically designed to handle time-varying information (i.e., navigation error and moving obstacles trajectories) while still exploiting the potential of "static" planners in giving optimized solutions. Specifically, static and dynamic information are handled by two different and sequential steps, defined as Step I and II. The two steps are repeated for each GNSS coverage map (j = 1,…,J) to associate a trajectory to each navigation level, thus avoiding the need to propagate the navigation error within the trajectory building process. At the end of step II, J polygonal chains are available as candidate solutions. Step III converts these trajectories in smoother paths and compares them, so that the trajectory which has the minimum cost is chosen as the final solution of the strategic planner. Details about the three steps are given in section from III.A to III.C.

A. Optimized Polygonal chain generation

Step I aims at defining for each GNSS map an optimized polygonal chain which accounts for all the static information. It is carried out by using the Open Motion Planning Library (OMPL) [37] framework, which implements some of the most common planners, and allows selecting the most suitable solution. A Batch informed tree (BIT*) [38] planner has been chosen for our aim, and modifications have been applied to the default version, to accommodate the multi-objective optimization and the multiple constraints derived in section II. BIT* is an informed sampling-based algorithm which performs "rewiring-and-cut" to the three to return an optimized solution with a reduced computational time. Because of the rewiring operation, no time dependent information and constraints can be accounted for during the solution building process. Indeed, rewiring changes locally the time history of the nodes, which would require checking again the path feasibility for all the following segments [39], [40].

The solution of each (j) iteration over GNSS coverage map is a polygonal chain minimizing the optimization cost while satisfying the planner bounds. The planner also reports, for each polygonal chain, segment information about mean velocity, time-to-fly and required energy to be covered either in wind (E_w) or no wind (E_0) condition. Planner segment information and planner costs and bounds are detailed in the following sub-sections.

1) Planner segment information

Planner segment is defined as the straight line included by two waypoints of the polygonal chain. Its mean velocity, elapsed time and mean energy depend on the wind direction and intensity, on the cruise velocity of the UAV, and on its aerodynamic characteristics. The ground velocity of the UAV in a North East Down (NED) frame, is referred to as $\mathbf{v}_g$ and is directed along the planner segment. The norm of this vector (i.e., v_g) is equal to the cruise velocity if allowed by the wind conditions. Since the segment could be in general very long while wind conditions can change with higher spatial frequency, v_g is defined by averaging on several subdivisions of the segment (referred to as *subsegments* in the following) with a spacing of Δs_v. For each (i) subsegment, the tentative airspeed ($\mathbf{v}_{a,i}$) is derived as the difference between the desired ground velocity (i.e., $\mathbf{v}_c$, directed along the segment and with norm equal to the cruise velocity, i.e., v_c) and the wind velocity sampled at the initial point of the subsegment, i.e., $\mathbf{v}_{w,i}$, as follows

$$\mathbf{v}_{a,i} = \mathbf{v}_c - \mathbf{v}_{w,i} \tag{2}$$

The ground velocity of the subsegment (i.e., $\mathbf{v}_{g,i}$) is given by

$$\mathbf{v}_{g,i} = \begin{cases} m_{-}\mathbf{v}_c & \left\|\mathbf{v}_{a,i}\right\| > v_{\max} \\ m_{+}\mathbf{v}_c & \mathbf{v}_{a,i}(s) < 0 \\ \mathbf{v}_c & otherwise \end{cases} \tag{3}$$

Where $\|\mathbf{a}\|$ and $\mathbf{a}(s)$ are the operators which give the norm of the vector and its projection along the segment direction, respectively. m_{-} is a constant which scales down the ground velocity in order to keep the airspeed equal to the 80% of the maximum admissible velocity by the vehicle ($v_{\max}$), in the case $\|\mathbf{v}_{a,i}\| > v_{\max}$. If velocity reduction brings the norm of ground velocity to be smaller than the minimum admissible velocity ($v_{\min}$) the path is considered not flyable. If the component of the airspeed along segment direction, i.e., $\mathbf{v}_{a,i}(s)$ is smaller than 0, the wind velocity on the vehicle tail is greater than the desired ground speed and it is convenient to increase the ground speed to accommodate the wind on the tail instead of counteracting it and thus producing an increase of energy consumption. Therefore the constraint m_{+} is used to make $\mathbf{v}_{a,i}(s) = 0$. Once $\mathbf{v}_{g,i}$ is obtained, the current airspeed (i.e., $\tilde{\mathbf{v}}_{a,i}$) is estimated by replacing $\mathbf{v}_c$ with $\mathbf{v}_{g,i}$ in (2). These velocities are used to compute energy consumption and define travelling time.

As in the velocity case, energy consumption and travelling time are defined in each subsegment (E_i and τ_i, respectively) and summed up to obtain the segment-related value. The total time for covering each subsegment is

$$\tau_i = \Delta s_v / v_{g,i} \,, \tag{4}$$

Energy consumption for electric multirotors is in general computed by considering the aerodynamic coefficients of both platform and propeller's blades, and using battery and motor models to retrieve the current intensity (with first order modeling), and the associated required power and energy [41], [42]. In this work, battery

and motor modelling are not accounted for, since an approximate estimate of energy consumption is required to verify path flyability. Furthermore, the involved modeling parameters may be hard to find for existing configurations, and the more complex model can affect the path definition by increasing the computational burden. Therefore, the energy consumption is obtained by multiplying τ_i by the required power of a flying helicopter (i.e., P_i) derived with the rotor theory [43], as

$$P_i = P_{r,i} + P_{0,i} + P_{p,i} + P_{c,i}\,, \tag{5}$$

where $P_{r,i}$ is the induced power obtained applying the rotor momentum theory, $P_{0,i}$ is the viscous power which depends on the rotor blade drag. $P_{p,i}$ and $P_{c,i}$ are the parasitic and climb power, required to overcome the platform drag and increase its gravitational potential, respectively.

Induced and parasite power can be expressed in terms of coefficients, i.e., $Cp_{r,i}$ and $Cp_{0,i}$, so that

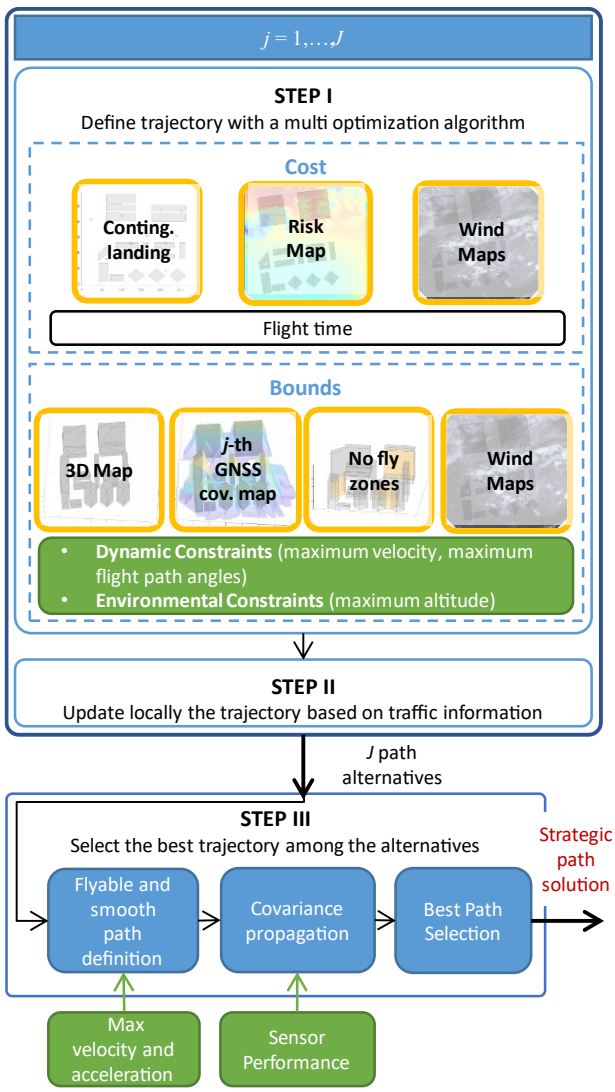


Fig. 2. Strategic planner flowchart

$$P_{r,i} + P_{0,i} = N_b \rho \pi r_b^2 \left(\Omega r_b\right)^3 \left(Cp_{r,i} + Cp_{0,i}\right). \tag{6}$$

where N_b, r_b and Ω are the number, the radius and the angular velocity of the propellers, respectively and ρ is the air density. $Cp_{r,i}$ is a function of the thrust coefficient (Ct):

$$Cp_{r,i} = \frac{\kappa Ct^{3/2}}{\sqrt{2}}\,. \tag{7}$$

κ is a coefficient accounting for non ideality losses which can be assumed equal to 1.15 [43]. Ct expression can be found in [42] for angular velocities of the blades expressed in RPM, using Ω in rad/s Ct is [44]:

$$Ct = \lambda \zeta^2 B_b \eta_0 \left(\frac{\varepsilon\theta_b - \alpha_b}{\pi AR_b + \eta_0}\right). \tag{8}$$

ε is the downwash factor, λ and ζ are correction factors. η_0 is a factor connecting the lift coefficient of the blade profile with its angle of attack. All the elements which are highlighted with a b subscript are blade-related, and include the blade number (B_b), its zero-lift angle (α_b) and aspect ratio (AR_b). θ_b is the angle of attack of the blade section, which, although counter intuitive, is constant independently from the forward velocity of the propeller, which is very small with respect to the rotational velocity of the blade. Conversely, the viscous power coefficient depends on the blade profile drag coefficient (i.e., C_d) [43]:

$$Cp_{0,i} = \frac{\sigma C_d}{8}\left(1 + K_v\left(\frac{\tilde{\mathbf{v}}_{a,i}(s)}{\Omega R_b^2}\right)^2\right). \tag{9}$$

where K_v is scaling factor accounting for forward velocity contribution to the profile drag and $\sigma = 2B_b/\pi AR_b$ is the blade solidity. C_d is constant and can be derived as detailed in [42], as a function of the zero drag coefficient.

The climb power exists only if a vertical velocity is envisaged and is equal to the product between this velocity and the vehicle weight. In descending phase, which requires less power than climbing, half of climbing power is accounted for.

The parasitic power contribution is given by computing the dot product between the velocity and drag vector, which depends on orientation of the vehicle. Vehicle orientation (pitch and roll angles) and required thrust (i.e., T_i) to bear the current airspeed, are found by solving the non-linear 3D trim equations, which are omitted here in for the sake of brevity. The interested reader is referred to [42]. Rotational velocity of the blades can be obtained by T_i using [44]:

$$\Omega = \sqrt{\frac{T_i / N_b}{\rho \pi R_b^4 Ct}}\,. \tag{10}$$

The adopted energy consumption model considers a quadratic dependance of drag on airspeed [45]. Multirotor UAS literature shows that linear drag modeling is effective

at low airspeeds and limited tilt angle [46]. Nevertheless, due to the main interest in moderate to relatively high airspeed, which also emphasizes the contribution of the drag generated by the frame (quadratic in nature), quadratic modeling is here considered adequate. Future works will address the injection in the planner of different energy consumption models and the effects on the final solution.

2) Planner Cost

The optimal path must be searched as the one minimizing the cost function which includes path length (l), ground risk (r), landing site (d) and energy/wind (e) costs.

$$f = w_l C_l + w_r C_r + w_d C_d + w_e C_e \,. \tag{11}$$

In (11) w_x indicates the weight factor for the cost C_x. The length cost is estimated by computing the polygonal chain extent, i.e., $C_l = L$, being L the overall path length. Risk and landing site costs are obtained by integrating over the path extension the risk and landing site information reported in the dimensionless maps shown in Fig. 1d and c, respectively. Using dimensionless information provides over the integration a quantity which has the same unit of the length cost. Following this approach, also wind/energy cost is obtained by integrating the energy information along the trajectory. In order to nondimensionalize this information, an approach similar to [22] has been used so that the energy contribution to cost is estimated by dividing the required energy with and without wind. For each *planner segment* of the energy cost is

$$\Delta C_e = \frac{E_w}{E_0}\Delta l = \frac{\sum_i P_i \Delta t_i}{\sum_i P_i\big|_{\|\mathbf{v}_{w,i}\|=0}\, \Delta t_i\big|_{\|\mathbf{v}_{w,i}\|=0}}\Delta l \,. \tag{12}$$

Where Δl is the segment length and the ratio presents at the numerator and denominator the energy obtained estimating P_i and Δt_i over each subsegment accounting or not for wind information, respectively. Specifically, the quantities $P_i\big|_{\|\mathbf{v}_{w,i}\|=0}$ and $\Delta t_i\big|_{\|\mathbf{v}_{w,i}\|=0}$ indicate subsegments' power and flying time in case no wind is considered. The overall energy cost i.e., C_e, is obtained by summing up ΔC_e related to each planner segment.

3) Planner Bounds

As far as the planner bounds are concerned, flyable trajectories are those which lie above the GNSS coverage map, are free from collisions with fixed obstacles and NFZ, and satisfy the altitude and flight angle limitations imposed by vehicle dynamic and/or airspace constraints. In addition, wind constraints are also accounted for, by excluding the paths which require to fly in excessive wind areas, or whose ground speed estimated with (3) is smaller than $v_{\min}$.

Collision free requirement is ensured by modeling the vehicle as a sphere of radius r_p, and avoiding its collisions with NFZ and fixed obstacles. The sphere radius is obtained by summing up a safety distance d_m to the maximum admissible positioning error ($\Delta p_{\max}$) which is scaled with a coefficient $K_m \geq 1$:

$$r_p = d_m + K_m \Delta p_{\max} \,. \tag{13}$$

Differently from [36], where the vehicle bound was modeled as an ellipsoid with time-changing size equal to the expected positioning error, in this work a worst case logic is applied and the maximum navigation error is accounted for to prevent vehicle collisions with obstacles. This choice allows using a time-invariant information which is compliant with the BIT* framework, and has little impact on the trajectory generation since the time varying safety ellipsoid plays a role only when $d_m = 0$, as demonstrated in [36].

Maximum endurance and energy consumption must also be taken into account when verifying path flyability. Total elapsed energy is estimated for each point of the BIT* tree by accounting for the previous history. Since rewiring operation changes the point history, cumulative energy information associated to tree's points cannot be used to exclude paths. Conversely, once the BIT* solution is available, energy consumption can be addressed and if the required energy is greater than the available one, cost steering mechanism is adopted to make the planner search for shorter paths, by increasing w_l in (11). If the BIT* planner returns an energy-unfeasible path, N_t tentatives are performed to obtain a feasible solution by incrementally increasing w_l.

B. Strategic Deconfliction

Time varying information about mobile obstacles, omitted in the previous step, is used at this stage to locally modify, when needed, the polynomial path resulting from STEP I. An RRT-based algorithm has been developed which aims at locally modifying the trajectory while driving the vehicle towards the previously computed trajectory when the collision has been solved, in order to keep trajectory optimality. RRT structure has been used instead of optimized RRT* [47] and its informed version [48], to store during the tree building process cumulative information about nodes, such as time of arrival and elapsed energy, without the need of recomputing it due to the rewiring process. A similar strategy has been adopted in [49], to accommodate energy and dynamic constraints when planning for the vehicle trajectory. The RRT-based local replanner uses the same bounds defined in STEP I to check path feasibility. In this case, the cumulative energy information is also taken into account to exclude non feasible points (i.e., whose energy consumption is higher than the maximum value). Velocity, energy and time required to fly each segment are derived as detailed in

section III.A.1. Landing site and risk map are not considered at this stage because they only contribute to the optimization cost which is not accounted for due to the non-optimal nature of the RRT algorithm.

The RRT building process, which is detailed in Algorithm 1, uses information about the list of waypoints obtained in the previous step, referred to as $\mathbf{W}$, to support informed sampling directed towards the previous planned path, and aims at building a feasible trajectory going from the first (i.e., $\mathbf{w}_{start}$) to the last (i.e., $\mathbf{w}_{goal}$) point of this list. The RRT tree, namely $\mathbf{\Gamma}$, stores for each node information about position, time of arrival, and total energy. The associated subtrees containing this information are referred to as $\mathbf{\Gamma}_p$, $\mathbf{\Gamma}_t$ and $\mathbf{\Gamma}_e$ respectively. $\mathbf{\Gamma}_p$ is initialized as $\mathbf{w}_{start}$, $\mathbf{\Gamma}_t$ as the initial time of the mission, i.e., t_0, and for the first node, the total energy is set equal to zero (lines 11-13). The process ends when $\mathbf{w}_{goal}$ is added to the tree (line 31) or the maximum number of iterations (i.e., K) is reached. At each step, the building process tries to add to the tree a point $\mathbf{w}_n$ from $\mathbf{W}/\mathbf{\Gamma}_p$ (line 16). This list includes the points belonging to $\mathbf{W}$ which have not been already added to $\mathbf{\Gamma}_p$. $\mathbf{w}_n$ is the first point in the $\mathbf{W}/\mathbf{\Gamma}_p$ list which satisfies

$$0 < \left(\mathbf{w}_{goal} - \mathbf{n}_{\mathbf{\Gamma}}\left(\mathbf{w}_{goal}\right)\right) \cdot \left(\mathbf{w}_n - \mathbf{n}_{\mathbf{\Gamma}}\left(\mathbf{w}_{goal}\right)\right) < 1, \qquad (14)$$

where $\mathbf{n}_{\mathbf{\Gamma}}()$ is the operator yielding the nearest neighbor within the tree of the points in the brackets. If the path to $\mathbf{w}_n$ from its nearest neighbor $\mathbf{n}_{\mathbf{\Gamma}}(\mathbf{w}_n)$ is feasible (line 17), $\mathbf{w}_n$ is selected as the new point to add to the tree, i.e., $\mathbf{x}_w$. If it is not, the last feasible point of the segment connecting $\mathbf{n}_{\mathbf{\Gamma}}(\mathbf{w}_n)$ to $\mathbf{w}_n$ (i.e., $\mathbf{x}_c$) is used as the central point to perform a random extraction from a gaussian distribution (lines 20-21). The random point $\mathbf{x}_r$ becomes the next candidate point to add to the tree (i.e., $\mathbf{x}_w$) if the segment going from $\mathbf{n}_{\mathbf{\Gamma}}(\mathbf{x}_r)$ to $\mathbf{x}_r$ is collision free (lines 22-24). If there is a suitable candidate to be added to the tree (line 26), $\mathbf{x}_w$ is added to the tree and connected to $\mathbf{x}_n$ which is either its nearest neighbor, i.e., $\mathbf{n}_{\mathbf{\Gamma}}(\mathbf{x}_w)$ or its preceding point in the tree if its path to $\mathbf{x}_w$ is feasible (line 27). The operation of selecting $\mathbf{x}_n$ by using as tentative the point preceding $\mathbf{n}_{\mathbf{\Gamma}}(\mathbf{x}_w)$ allows neglecting nodes which can make the path unnecessarily long without performing further trajectory pruning after path building, which is not allowed because of the time dependence of the nodes (that, if altered, could compromise the path safety). Once $\mathbf{x}_w$ and $\mathbf{x}_n$ are available, the position tree is updated along with the energy and time trees, knowing the mean velocity and total energy required to cover the segment from $\mathbf{x}_n$ to $\mathbf{x}_w$ (line 28). After tree update the path from $\mathbf{x}_w$ to the point following $\mathbf{w}_n$ in $\mathbf{W}$, i.e., $\mathbf{w}_{n+}$, is checked for feasibility and if verified, $\mathbf{w}_{n+}$ is added to the tree (lines 29-33). This condition allows adding $\mathbf{w}_{goal}$ after $\mathbf{x}_w$ if the latter is close to the end of the former polygonal chain, thus avoiding random sample to iteratively pick a point near to the goal to end the process.

Algorithm 1

```
1  Input:  w_start, w_goal// Start and end waypoints
2          t_0     // Initial time
3          W       // Step I waypoint list
4          α       // Maximum flight path angle
5          h_lim   // Altitude limits
6          ξ       // Maximum battery capacity
7          M       // occupancy map (fixed & mobile obstacle, NFZ)
8          D_j     // GNSS coverage map
9          Z       // Wind map
10         K       // maximum iteration
11 Γ_p             ← {w_s}
12 Γ_t             ← {t_0}
13 Γ_e             ← {0}
14 for (k=0; k < K; k++)
15     x_w         ← {∅}
16     w_n         ← GET_NEXT_POINT(W, Γ_p, w_goal)
17     if IS_FEASIBLE(n(w_n),w_n,M,Z,D_j,ξ,α,h_lim)
18         x_w  ←  w_n
19     else
20         x_c  ←  GET_LAST_FEASIBLE(n(w_n),w_n,M,Z,D_j,ξ,α,h_lim)
21         x_r  ←  RANDOM_SAMPLE(x_l)
22         if  IS_FEASIBLE(n(x_r), x_r,M,Z,D_j,ξ,α,h_lim)
23             x_w  ←  x_r
24         end if
25     end if
26     if x_w ≠ ∅
27         x_n ←  GET_CONNECTING_POINT(x_w,M,Z,D_j,ξ,α,h_lim)
28         (Γ_p, Γ_t, Γ_e) ← UPDATE_TREE (Γ_p, Γ_t, Γ_e, x_w, x_n)
29         if IS_FEASIBLE(x_w, w_n+,M,Z,D_j,ξ,α,h_lim)
30             (Γ_p, Γ_t, Γ_e) ← UPDATE_TREE (Γ_p, Γ_t, Γ_e, w_n+, x_w)
31             if  w_n+= w_goal, return
32         end if
33     end if
34 end for
35 Return:  E(Γ_p)  // polygonal chain connecting w_start and w_goal
36         E(Γ_t),E(Γ_e)// elapsed time and energy of E(Γ_p)
```

C. Optimal trajectory selection

The final step compares the J trajectories generated for each DOP level by the previous two steps. As reported in Fig. 2, the J polygonal chains are first transformed into smooth and flyable polynomial trajectories [50], along which the navigation error covariance is propagated by using an extended Kalman filter (EKF) exploiting GNSS pseudoranges and inertial measurements in tightly coupled integration. The reader is referred to [5] for the detailed formulation of the EKF. Only GNSS signals whose raypath is not obstructed by fixed obstacles are considered for covariance propagation. To predict GNSS constellation geometry, the information of t_0 in UTC is required.

The standard deviation of the positioning error along the trajectory (σ_p) is compared to the maximum allowed positioning error, i.e., Δp_{max}. The trajectories for which $\sigma_p > \Delta p_{max}$ in at least one point of the time history are deemed not feasible and removed from the candidate list. The remaining trajectories are compared based on their cost f, which is computed again along the polynomial trajectory which can slightly deviate from its associated polygonal chain thus providing cost variation. In addition, since polynomial trajectory does not fulfill the constant velocity hypothesis which holds true in each segment of the polygonal chain, costs are estimated by integrating the risk,

energy and landing information over time instead of path length thus modifying (11) as follows

$$f = w_l L + \frac{w_r \int_{t_0}^{t_f} \beta_r(t)\,dt + w_d \int_{t_0}^{t_f} \beta_d(t)\,dt + w_e \int_{t_0}^{t_f} \beta_e(t)\,dt}{t_f - t_0} L \quad (15)$$

where L is the trajectory length and t_0 and t_f are the start and final time of the trajectory, respectively. $\beta_r(t)$, $\beta_d(t)$ and $\beta_e(t)$ are the punctual information about risk, landing site and energy which can be obtained from risk and landing site maps, or by estimating the ratio between windy and non windy required energy along the trajectory. The time integral (dt) is more indicative of the effective cost when a non constant velocity is assumed.

IV. RESULTS

The path planning algorithm performance is tested in a relevant environment, taken from a real-world scenario and consisting in a portion of the Naples city center. A portion of the city has been selected which is characterized by high buildings where navigation issues can arise. The associated 3D obstacles and terrain are reported in Fig. 3. Two pairs of start and goal points have been considered to test the planner and are reported with circles (goal points) or crosses (start points) in the figure. The two test cases are defined as follows:

Case 1. Start point in [-689 302 30] and goal point in [503 117 20], reported in red in Fig. 3.

Case 2. Start point in [-1273 797 40] and goal point in [1111 -470 20], reported in green in Fig. 3.

It is clear that since Case 2 points are more distant than those used in Case 1, a longer trajectory is expected.

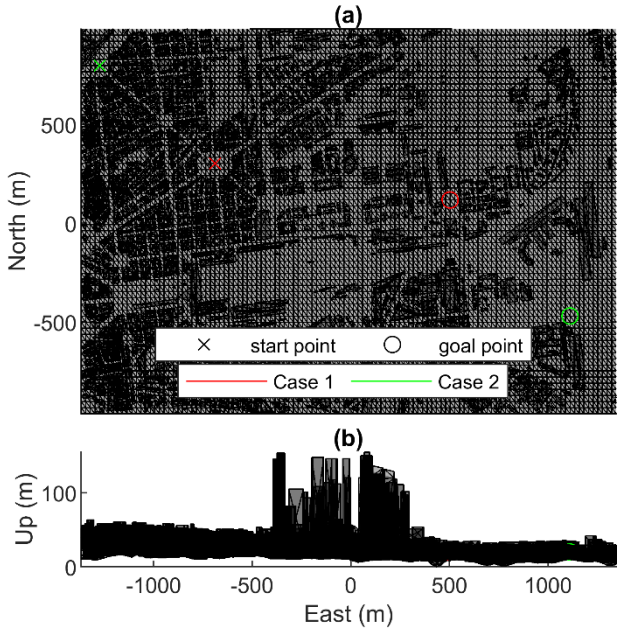


Fig. 3. Planning scenario, (a) top and (b) lateral view.

NFZ play the same role of obstacles, therefore they have not been considered for planning performance evaluation. Conversely, the top views of the 2D maps such as risk, landing site and wind maps are reported in Fig. 4. Risk map associates the highest risk to the train station (highlighted by the yellow background). A lower risk is obtained over buildings. Wind map foresees a constant wind directed towards East of intensity 5 m/s and zero elevation. To account for navigation issues, three GNSS coverage maps have been derived with a resolution of 5 m. They are associated with increasing level of DOP (2, 3, and 4), which reduces their elevation as remarked in Fig. 5. For the sake of visualization, figures from Fig. 5a to Fig. 5c show the top view of the GNSS maps in a reduced portion of the environment characterized by high buildings where non nominal GNSS conditions could arise.

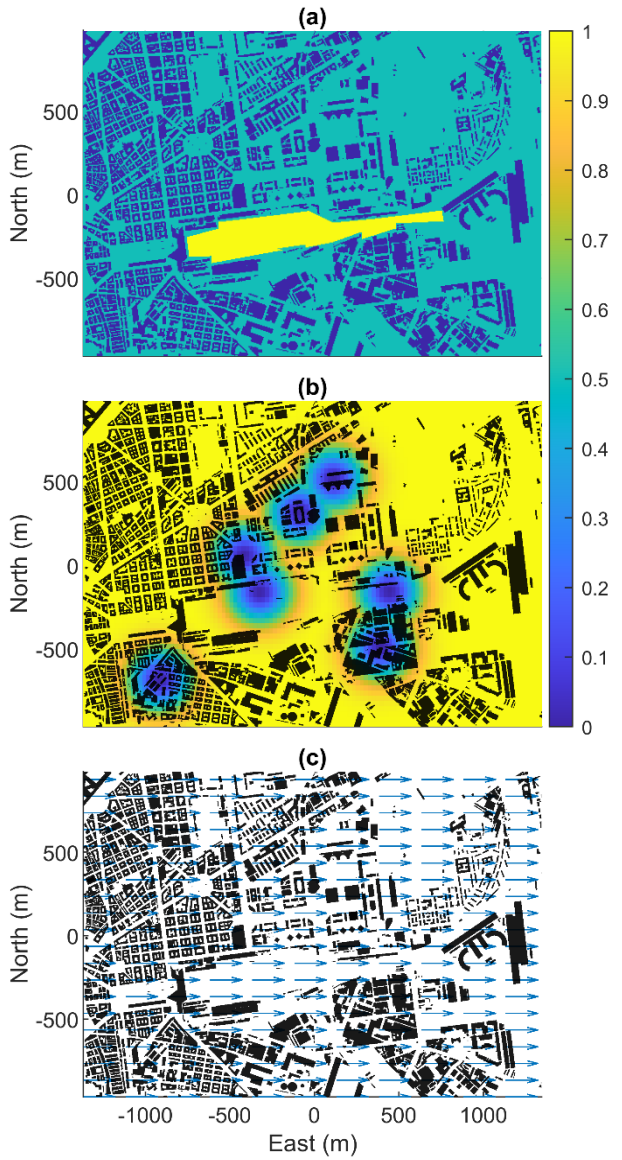


Fig. 4. Planning scenario, (a) risk (b) landing and (c) wind map, top view. Maps (b) and (c) also report in black the top view of the obstacle to give context to the maps

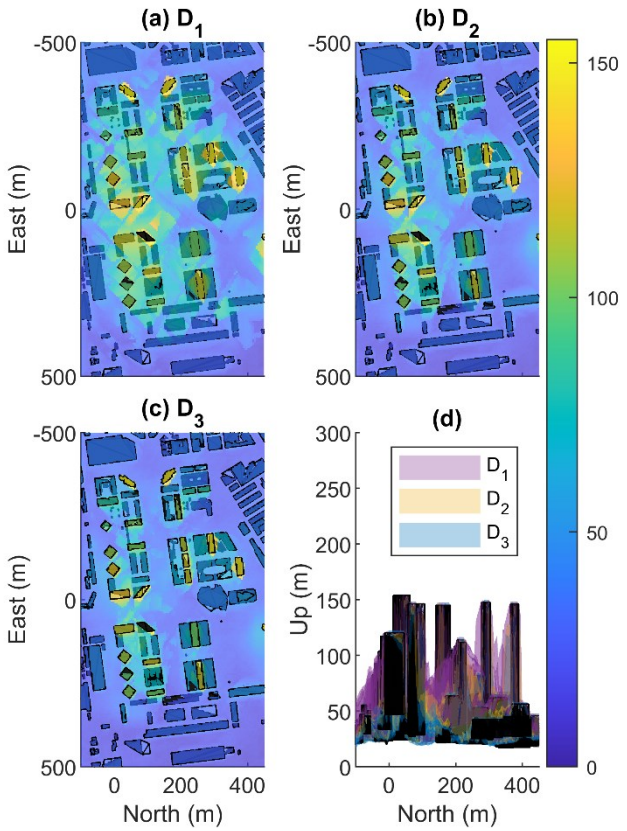


Fig. 5. GNSS Coverage maps associated to DOP equal to (a) 2.0, (b) 3.0 (c) 4.0. Figures (a) from (c) reports the topview of GNSS coverage map, whose altitude in meters is reported by the colorbar. (d) Comparison of the three maps, lateral view.

The three GNSS coverage maps are reported with three different colors in Fig. 5d, to better remark their different spatial extension.

The aircraft considered for the simulation is the DJI M300 RTK whose assumed specifications and constraints are reported in Table I. Navigation performance of the IMU has also been included and is assumed to be the one of the medium grade IMU HG1120AA50 from Honeywell [51]. Only accelerometer specifics are reported, because within the assumed covariance propagation scheme [5] they are the only parameters impacting positioning error propagation.

To highlight the ability of the conceived framework to deal with several constraints, while shaping its final result as a function of inputs and weighting coefficients of the cost function, the results of the planner while varying these parameters are shown in section IV.A. Here, it is assumed that the start and the end points are those of case 1 and no time-varying information is available, so that STEP II does not have to modify STEP I trajectories. Conversely, sub-section IV.B analyzes the planner behavior including traffic information and thus strategic deconfliction through STEP II. The results of all the three steps are reported in this subsection while using as reference case 2 start and end points. It is worth noting that being both STEP I and STEP II characterized by heuristic processes, the planner solution is non-deterministic.

TABLE I
VEHICLE SPECIFICS AND AIRSPACE CONSTRAINTS

Constraints		Value
Battery capacity	ξ (mAh)	11870
Maximum airspeed	(m/s)	23
Max wind speed	(m/s)	15
Cruise speed	v_c (m/s)	10
Maximum FPA	α (°)	15
Max Positioning error	Δp_{max} (m)	2
Safety margins	d_m (m)	3
	K_m	1
IMU Parameters	Acc. In-run stability (mg)	0.20
	Velocity random walk (m/s/$\sqrt{h}$)	0.15
Altitude limits	Min/Max altitude (m)	0-80

A. Inputs and weighting coefficients variation

This section underlines how STEP I output (and therefore planning output) modifies as a function of the coefficients of the cost function (in section IV.A.1), and planning inputs, considering Case 1. Specifically, variation of wind map and ground cost map (i.e., risk and/or landing site map) are tackled in sections IV.A.2 and IV.A.3, respectively. Finally, the impact of GNSS coverage map on navigation performance is discussed in section IV.A.4. Since the first three sub-sections are not discussing navigation performance, they all assume D_3 (see Fig. 5c) as reference for GNSS coverage map.

1) Weighting coefficient variation

The effect of the weighting factors in shaping the UAV trajectory is highlighted in Fig. 6. The figure is reported to outline what is the behavior of the planner when the path selection either is driven by only one cost contribution (setting the other weighting coefficients to zero) or mixes all of them. To this aim, figures from Fig. 6a to Fig. 6d show the results obtained by setting to zero all the weighting coefficients except one which is length, risk, landing, or wind cost, respectively. Conversely, Fig. 6e shows the solution when setting $w_r = w_d = 2$ and $w_l = w_e = 1$. Associated cost maps to each case are reported in the figure along with 3D obstacles, for the sake of visualization. The shortest trajectory compliant with UAV constraints is reported in Fig. 6a. Fig. 6b reports the optimal trajectory in terms of ground risk, which mainly flies over buildings. In the case the main priority of the planner is guaranteeing a "landing safe" trajectory, the path deviates so to mostly cover the blue area of landing site map, as shown in Fig. 6c. Conversely a trajectory which is mostly headed toward east (which is parallel to the wind direction) is obtained when considering all weighting coefficients equal to zero except w_e, as reported in Fig. 6d. Fig. 6e shows the trajectory obtained when mixing the cost contribution. In this case, the optimization cost associated is mostly focused on minimizing risk and landing cost with respect to the other two. As expected, the resulting

trajectory is closer to the landing safe path reported in Fig. 6c, while also complying with other costs requirements. Specifically, deviations from landing safe trajectory are experienced to push away the path from the riskiest area (yellow in Fig. 6b). The costs obtained in the 5 analyzed configurations are reported in Table II. The table highlights that the landing cost of the case reported in Fig. 6e, slightly increases with respect to the one of trajectory aimed at landing site cost minimization. Conversely, when aiming at optimizing all the cost contributions, all C_l, C_e and C_r reduce. However, being the risk weight higher than the others, a dramatic reduction of C_r is experienced.

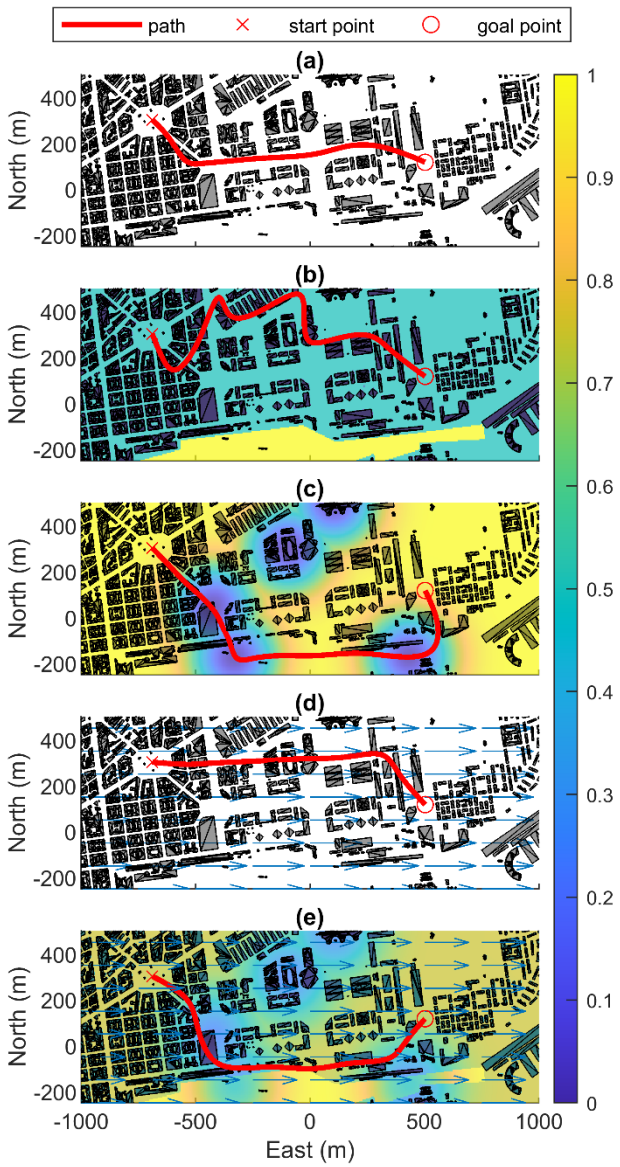


Fig. 6. Path solution of strategic planner in case 1. (a) $w_l = 1$, $w_r = w_d = w_e = 0$; (b) $w_r = 1$, $w_d = w_l = w_e = 0$ and dimensionless risk map; (c) $w_d = 1$, $w_l = w_r = w_e = 0$ and dimensionless landing site map; (d) $w_e = 1$, $w_l = w_r = w_d = 0$ and wind map; (e) $w_l = w_e = 1$, $w_r = w_d = 2$, wind map and dimensionless sum of risk and landing site maps. D_3 GNSS coverage map is used to define the area the UAV cannot cross due to navigation issue.

TABLE II
TRAJECTORY COSTS BY CHANGING WEIGHTING COEFFICIENTS, CASE 1

Weighting coefficients	Fig	Cost functions (m)				
		C_l	C_r	C_s	C_e	f
$w_l = 1$, $w_r = w_d = w_e = 0$	6a	1314.9	518.3	895.4	1234.9	1314.9
$w_r = 1$, $w_l = w_d = w_e = 0$	6b	1533.7	479.2	1098.5	1431.5	479.2
$w_d = 1$, $w_l = w_r = w_e = 0$	6c	1764.2	1138.8	851.1	1670.9	851.1
$w_e = 1$, $w_l = w_r = w_d = 0$	6d	1318.5	536.7	896.3	1228.2	1318.5
$w_l = w_e = 1$, $w_r = w_d = 2$	6e	1507.5	502.7	902.3	1416.2	5733.8

2) *Wind variation*

If the wind direction varies with respect to the one reported in Fig. 5c, the path will be modified as well to meet energy cost minimization, while trying to be parallel to the wind direction. Fig. 7 reports the path obtained if the wind direction has a 120° angle with respect to the North direction. To emphasize the effect of wind variation, the planner is run assuming $w_e = 1$, $w_l = w_r = w_d = 0$, resulting in an overall cost of 1262.7 m. Differently from the trajectory reported in Fig. 6d, the trajectory is mostly parallel to the 120° direction remarked by the wind vector in the figure.

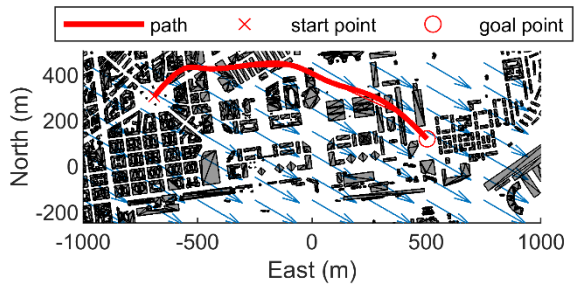


Fig. 7. Path solution of strategic planner over case 1 request, obtained by setting $w_e = 1$, $w_d = w_r = w_l = 0$, and wind map used to obtain the path. Wind is uniform in the scenario and has zero elevation and 120° azimuth. D_3 GNSS coverage map is used to delineate the area the UAV cannot cross due to navigation issue.

3) *Ground cost variation*

The performance of the planner while changing ground cost (which combines landing sites and risk) is assessed by accounting for only one of the ground information sources, for the sake of brevity. To this aim, landing site location variation is considered. The landing sites are moved with respect to those reported in Fig. 5b, so that they are placed in the central area of the Naples business center which is characterized by very high buildings. The novel landing site map is reported in Fig. 8 along with the trajectory obtained setting $w_d = 1$, $w_l = w_r = w_e = 0$. The figure highlights how the planner is able to optimize the cost function by deviating its trajectory so as to cover the safer area for landing (reported in blue), obtaining a final cost of 550.0 m.

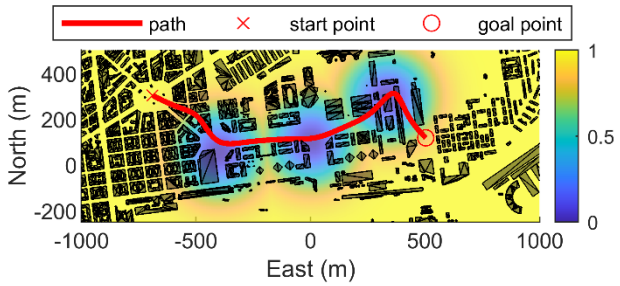


Fig. 8. Path solution of strategic planner over case 1 request, obtained by setting $w_d = 1$, $w_l = w_r = w_e = 0$, and nondimensional landing site map used to obtain the path. D_3 GNSS coverage map is used to delineate the area the UAV cannot cross due to navigation issue.

4) Navigation performance variation

Using the same landing site map and weighting coefficients of section IV.A.3 (i.e., $w_d = 1$, $w_l = w_r = w_e = 0$), Fig. 9 shows how the planned trajectory is modified when changing the GNSS coverage map. The top view of the paths obtained for each DOP level and GNSS coverage maps are reported in Fig. 9a. The area included within the D_1 map is also highlighted in gray. Although D_2 and D_3 paths are very close to each other and aimed at covering the area minimizing the landing cost, the extension of the D_1 map does not allow its associated trajectory to enter there. Indeed, paths populating that area fall below the D_1 map as remarked by the gray shadow covering portion of D_2 and D_3 paths. As the GNSS map elevation reduces by going from D_1 to D_3, a huge free space is available for the UAV motion, which is able to fly over a path characterized by a lower cost function. A cost of 932.2 m can be obtained when D_1 is used with respect to D_2 (665.0 m) and D_3 (558.0 m) cases.

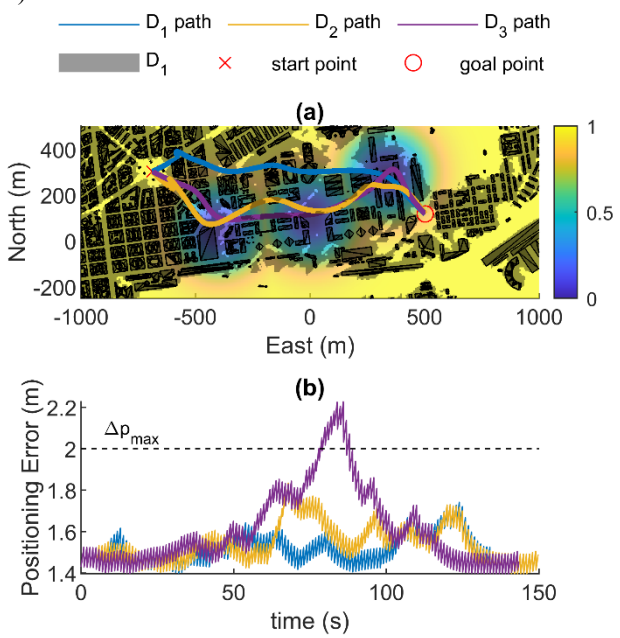


Fig. 9. Path solution of strategic planner over case 1 request, obtained by setting $w_d = 1$, $w_l = w_r = w_e = 0$, and using different GNSS coverage maps. (a) Obtained paths and used nondimensional landing site map. (b) Positioning error of the UAV over the three paths.

The trajectory associated with the highest DOP level easily falls below the GNSS coverage maps of the lower DOP indices (i.e., D_1), resulting in possible navigation error degradation, which only UAV equipped with high performance IMU (or with other A-PNT solutions) can tackle. Fig. 9b shows the positioning error obtained with an UAV embarking the HG1120AA50 IMU. The D_3 trajectory, which violates both the GNSS coverage maps with $j<3$, is the most critical and experiences a significant positioning error, which exceed Δp_{max}, thus it cannot be flown by the vehicle. This also highlights the technology-aware nature of the planning framework.

B. Strategic planning solution

This section reports the solution of the entire strategic planner pipeline considering case 2. Inputs included in Fig. 4 have been used as risk, landing and wind information. The planner has been set to consider the risk cost as the highest one so that $w_l = w_e = 1$, $w_r = 4$, $w_d = 2$. Time varying information is stored within the planner using map extension, i.e., Δt = 20 s, and Δt_c = 10 s, d_c = 10 m as margins for the intruders. The polygonal chains derived by STEP I are reported in Fig. 10. Due to the high-risk weight, all the trajectories aim at deviating in the bottom part of the figure, in order to avoid flying over the highest risk area. It can be observed that the trajectory associated to D_3 is the shortest one due to the smaller elevation of its associated GNSS coverage map. Four intruders have been selected, whose trajectories, reported in Fig. 10 have been designed to intersect several times the paths retrieved in STEP I.

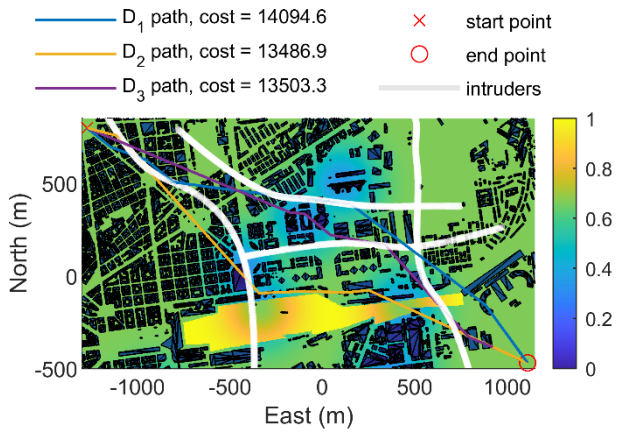


Fig. 10. STEP I polygonal chain associated to each GNSS coverage map over case 2 request. Adimensional sum of risk and landing site maps is reported as background color. Used inputs for wind, risk and landing maps are reported in Fig. 4. $w_l = w_e = 1$, $w_r = 4$, $w_d = 2$. Intruders' trajectories are reported in white.

The resulting polygonal chains obtained after performing mobile obstacle deconfliction with STEP II are reported from Fig. 11a to Fig. 11c, for each GNSS coverage map, as well as the polynomial trajectory derived in STEP III. The polygonal chains of STEP II slightly deviate from the STEP I output, remarking the effectiveness of the

modified RRT planner in providing local variations to the path, thus preserving or only slightly modifying its cost. For the sake of completeness, the costs of each path estimated with and without obstacles are reported in Table III. The former is derived by using the polynomial paths reported in Fig. 11. Conversely, the cost associated with no intruders are obtained by converting the polygonal chains of STEP I in polynomial smooth paths, and are slightly different than those reported in Fig. 10. Navigation Error along the three polynomial paths using the IMU whose parameters are reported in Table I, is shown in Fig. 12. Each trajectory is feasible, and thus a candidate solution for STEP III, because its navigation error is always smaller than Δp_{max}. It can be seen that, as in the previous section, D_3 trajectory is the one achieving the maximum navigation error because it can enter in GNSS-challenging areas due to the lower elevation of its associated GNSS map. The solution of the STEP III is selected among the candidates as the one with the minimum cost, thus being D_2.

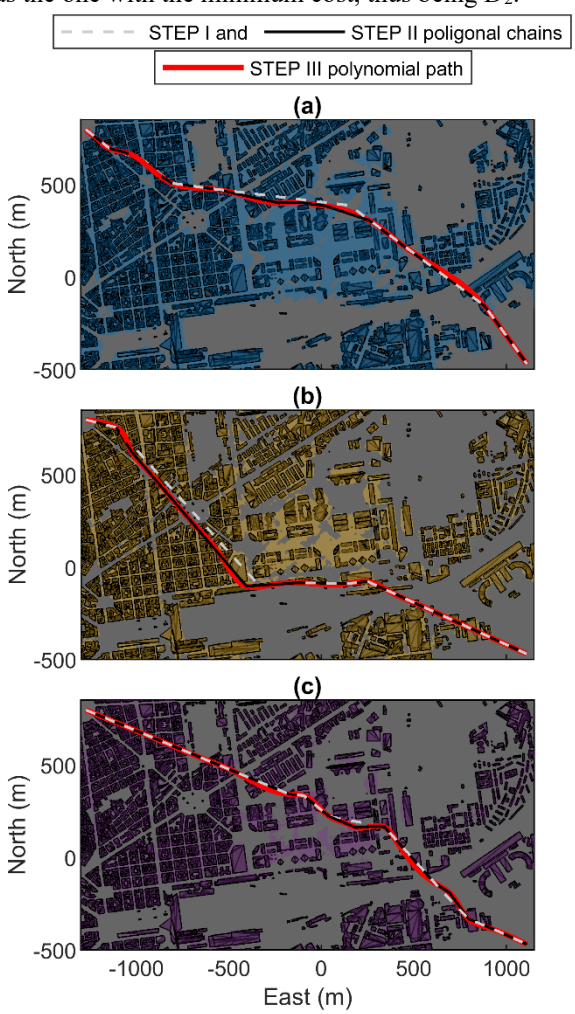


Fig. 11. Case 2 solutions associated to (a) D_1, (b) D_2, (c) D_3 GNSS coverage maps. STEP I, STEP II and polynomial paths are reported. Background shadow underlines the GNSS coverage map associated to each path. Used inputs for wind, risk and landing maps are reported in Fig. 4. $w_l = w_e = 1$, $w_r = 4$, $w_d = 2$. $\Delta t_c = 10$ s, $d_c = 10$ m.

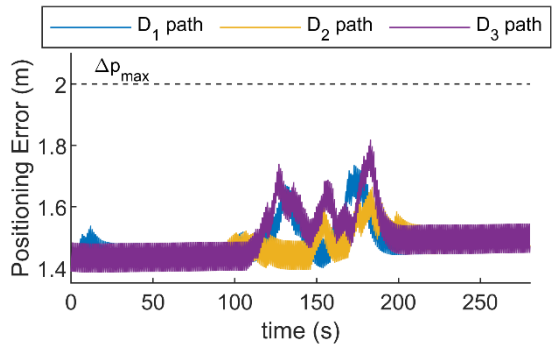


Fig. 12. Navigation error along the polynomial paths reported in Fig. 11.

TABLE III
COST OF EACH TRAJECTORY WITH AND WITHOUT OBSTACLES, CASE 2.

GNSS Maps	With intruders	Cost functions (m)				
		C_l	C_r	C_s	C_e	f
D_1	No	2838.5	970.2	2466.8	2667.0	14320.0
	Yes	2831.8	952.9	2464.0	2662.7	14234.2
D_2	No	2878.8	910.5	2100.7	2708.5	13230.9
	Yes	2909.5	950.0	2142.3	2735.6	13730.0
D_3	No	2807.7	914.6	2328.3	2628.0	13750.7
	Yes	2816.9	940.7	2296.0	2649.2	13820.8

1) Intruder safety margin variation

To further assess the performance of STEP II, this section analyses the variation on its solution when Δt_c and d_c are increased. Specifically, information about polynomial paths obtained by increasing Δt_c (i.e., $\Delta t_c = 20$ s) and leaving unaltered d_c and their associated costs are reported in Fig. 13 and Table IV respectively. Only STEP III polynomial paths are reported in the figure, where both top view and lateral view are shown. Conversely, the results obtained in case d_c changes and $\Delta t_c = 10$ s, are reported in Table V. Due to the heuristic nature of the STEP II, the same seed has been used in all the simulations to have comparable results. Results show that increasing intruder margins generates more conflicts and anticipates the already existing ones, leading to an increased path length. Trajectory deviation is occurring on the horizontal or on the vertical plane. The latter behavior is shown by the D_2 or D_3 paths in Fig. 13a.

In the case the spatial margin of the obstacle is enlarged, D_2 trajectory shows a reduction of path length. Indeed, obstacle enlarging produces a huge region where the STEP I path is in conflict with the intruder, wider than the one obtained in the smallest d_c case (i.e., $d_c = 10$ m).

TABLE IV
POLYNOMIAL PATH COSTS, WHILE CHANGING Δt_c, $d_c = 10$ m.

GNSS Maps	Δt_c (s)	Cost functions (m)				
		C_l	C_r	C_s	C_e	f
D_1	10	2831.8	952.9	2464.0	2662.7	14234.2
	20	2839.7	972.3	2454.0	2667.9	14304.9
D_2	10	2909.5	950.0	2142.3	2735.6	13730.0
	20	2910.8	944.1	2144.7	2736.9	13713.5
D_3	10	2816.9	940.7	2296.0	2649.2	13820.8
	20	2806.1	870.0	2301.2	2636.9	13525.4

TABLE V
POLYNOMIAL PATH COSTS, WHILE CHANGING d_c, Δt_c = 10 s.

GNSS Maps	d_c (m)	Cost functions (m)				
		C_l	C_r	C_s	C_e	f
D_1	10	2831.8	952.9	2464.0	2662.7	14234.2
	15	2844.7	987.7	2454.2	2673.6	14377.4
	20	2953.9	1011.1	2569.3	2773.8	14910.9
D_2	10	2909.5	950.0	2142.3	2735.6	13730.0
	15	2877.0	937.2	2122.5	2702.4	13578.4
	20	2847.1	984.6	2125.5	2672.6	13709.3
D_3	10	2816.9	940.7	2296.0	2649.2	13820.8
	15	2854.0	909.3	2323.7	2671.8	13810.3
	20	2890.5	958.2	2362.8	2689.0	14137.8

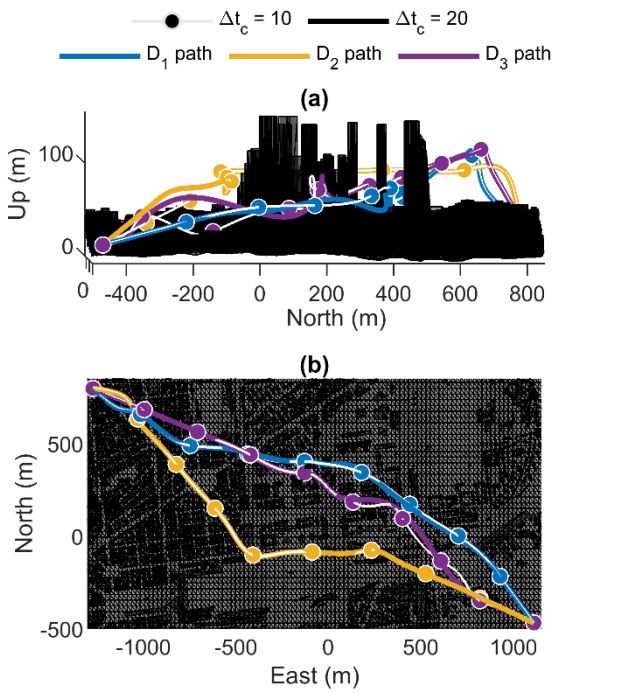


Fig. 13. Case 2 polynomial path obtained with either Δt_c = 10 s, d_c = 10 m, or Δt_c = 20 s, d_c = 10 m. Solution for all GNSS coverage maps are displayed. Used inputs for wind, risk and landing maps are reported in Fig. 4. $w_l = w_e = 1$, $w_r = 4$, $w_d = 2$. (a) lateral and (b) top view.

This requires the STEP II to plan for a path that highly deviates from STEP I polygonal chain, which finally produces, for the selected case, a reduced path length.

2) Computational time

The computational time required to solve each STEP I solution is set by the user in the OMPL framework, indeed BIT* optimization runs until a timeout which has been set to 10 seconds in this specific case. Therefore, it does not strictly depend on the distance between the start and the goal point and can be set by the user trade-offing optimality (which requires computational time increase) and time-to-solution. Is however clear that increasing scenario's dimensions and complexity will penalize the solution optimality for a given timeout. As far as STEP II is concerned, computational time does mostly depend on the airspace density, and thus on the number and the dimension of obstacles the STEP I trajectories are in collision with (i.e., Δt_c and d_c). Results in terms of computational time obtained as a mean of 30 runs performed with an intel i7 PC running MATLAB on Windows operating system at 2.60 GHz are reported in table VI, by using the same sets of Δt_c and d_c reported in the previous section. It can be seen that computational time mostly increases with d_c. Conversely, Δt_c increase slightly affects the computational time. Finally, STEP III aims only at comparing the solutions and is thus very fast (in the order of 0.5 s). The final strategic solution can thus be obtained in about one minute in the considered case (i.e., 2.7 km distance between start and goal point), which is compliant with the off-line requirement of the conceived planning framework.

TABLE VI
STEP II COMPUTATIONAL TIME.

Δt_c (s)	d_c (m)	Mean Computational time (s)		
		D_1	D_2	D_3
10	10	4.55	5.77	1.60
	15	7.62	6.97	2.74
	20	11.04	7.16	5.74
20	10	5.02	5.79	1.82

V. CONCLUSION

This paper introduced a planning framework for rotorcraft flying in an unstructured urban airspace. The ability of the planner to deal with a significant amount of information (either static or time varying) is ensured by information synthesis and sequential trajectory building and refinement, carried out through STEP I and II. Handling in two separate steps static and time varying information allows guaranteeing that cost-effective solutions are resulting from STEP I. In addition, the usage of GNSS maps, further reduces the computational effort by preventing computationally slow information propagation (i.e., navigation state covariance) to be performed within both path building and path refinement steps. Dynamic constraints and maximum endurance of the vehicle are also accounted for during path construction. Energy consumption of the UAV has been derived with a simplified formula with respect to the one exploited in the open literature, but still using physical characteristics and geometric factor of the UAV, thus being potentially ready to be extended to several types of multirotor UAVs.

Results demonstrate the effectiveness of the STEP I solution in dealing with multi-objective optimization and effectively shaping the trajectory geometry based on the weighting coefficients. The advantage of the RRT-based STEP II algorithm has been also investigated, by highlighting the local deviation property which allow preserving the cost of STEP I trajectory while still performing collision deconfliction with mobile obstacles. Finally, using GNSS maps instead of propagating navigation error along trajectory building generates several candidate solutions. This provides alternatives for the path solution without requiring the path planner to run several

times. The multi-constrained and multi-optimization nature of the planner, coupled with its modularity and low computational burden suggest this solution could be used as core algorithmic block for low altitude airspace management in Urban Air Mobility.

Further works are also aimed at integrating the proposed methodology and developed information handling approaches within structured environments, with the aim of comparing the results of the current paper with those obtained within a structured airspace scenario. In addition, further updates of the planning algorithms could be envisaged so as to adapt to the technological improvements in maps information gathering, e.g., in terms of more detailed and 3D varying wind maps. Injecting more accurate wind information would allow reducing the margins relevant to the endurance, thus increasing the efficiency of flight operations.

REFERENCES

[1] A. Bauranov and J. Rakas, "Designing airspace for urban air mobility: A review of concepts and approaches," *Prog. Aerosp. Sci.*, vol. 125, p. 100726, 2021, doi: https://doi.org/10.1016/j.paerosci.2021.100726.

[2] A. Chakrabarty, V. Stepanyan, K. Krishnakumar, and C. Ippolito, "Real-time path planning for multi-copters flying in UTM-TCL4," in *AIAA Scitech 2019 Forum*, San Diego, California, 2019.

[3] Spirent, "GNSS Foresight for aviation," 2022. https://www.spirent.com/assets/u/gnss-foresight-for-aviation (accessed Jun. 28, 2022).

[4] C. Struempfel, E. Schuster, S. Huschbeck, C. Berth, and M. Uijt De Haag, "Assured Multi-Mode Navigation for Urban Operations of Small UAS," in *AIAA Scitech 2020 Forum*, American Institute of Aeronautics and Astronautics, 2020.

[5] F. Causa and G. Fasano, "Multiple UAVs trajectory generation and waypoint assignment in urban environment based on DOP maps," *Aerosp. Sci. Technol.*, vol. 110, p. 106507, 2021, doi: https://doi.org/10.1016/j.ast.2021.106507.

[6] A. la Cour-Harbo, "Quantifying Risk of Ground Impact Fatalities for Small Unmanned Aircraft," *J. Intell. Robot. Syst.*, vol. 93, no. 1, pp. 367–384, 2019, doi: 10.1007/s10846-018-0853-1.

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

[8] S. Primatesta, A. Rizzo, and A. la Cour-Harbo, "Ground Risk Map for Unmanned Aircraft in Urban Environments," *J. Intell. Robot. Syst.*, vol. 97, no. 3, pp. 489–509, 2020, doi: 10.1007/s10846-019-01015-z.

[9] K. A. Adkins, M. Akbas, and M. Compere, "Real-Time Urban Weather Observations for Urban Air Mobility," *Int. J. Aviat. Aeronaut. Aerosp.*, vol. 7, no. 4, pp. 1–27, 2020.

[10] C. Ebert, C. Ruwisch, J. Weiss, M. Uijt De Haag, and F. Silvestre, "Trajectory Planning in Windy Urban Environment - a Gappy POD Approach for Wind Field Estimates with Sparse Sensors," in *AIAA AVIATION 2022 Forum*, American Institute of Aeronautics and Astronautics, 2022.

[11] E. Sunil *et al.*, "Metropolis: Relating Airspace Structure and Capacity for Extreme Traffic Densities," in *11th USA/EUROPE Air Traffic Management R&D Seminar*, Lisboa, Portugal, 2015.

[12] Y. Tang, Y. Xu, and G. Inalhan, "An Integrated Approach for On-Demand Dynamic Capacity Management Service in U-space," *IEEE Trans. Aerosp. Electron. Syst.*, p. 1, 2022, doi: 10.1109/TAES.2022.3159317.

[13] P. F. A. Di Donato and E. M. Atkins, "Evaluating Risk to People and Property for Aircraft Emergency Landing Planning," *J. Aerosp. Inf. Syst.*, vol. 14, no. 5, pp. 259–278, Apr. 2017, doi: 10.2514/1.I010513.

[14] C. A. Ochoa and E. M. Atkins, "Fail-Safe Navigation for Autonomous Urban Multicopter Flight," in *AIAA Information Systems-AIAA Infotech @ Aerospace*, American Institute of Aeronautics and Astronautics, 2017.

[15] M. Ortlieb, F.-M. Adolf, and F. Holzapfel, "Computation of a Database of Trajectories and Primitives for Decision-Based Contingency Management of UAVs over Congested Areas," in *2021 IEEE/AIAA 40th Digital Avionics Systems Conference (DASC)*, 2021, pp. 1–8, doi: 10.1109/DASC52595.2021.9594333.

[16] J. Sláma, P. Váňa, and J. Faigl, "Risk-aware Trajectory Planning in Urban Environments with Safe Emergency Landing Guarantee," in *2021 IEEE 17th International Conference on Automation Science and Engineering (CASE)*, 2021, pp. 1606–1612, doi: 10.1109/CASE49439.2021.9551407.

[17] S. Primatesta, G. Guglieri, and A. Rizzo, "A Risk-Aware Path Planning Strategy for UAVs in Urban Environments," *J. Intell. Robot. Syst.*, vol. 95, no. 2, pp. 629–643, 2019, doi: 10.1007/s10846-018-0924-3.

[18] C. A. Ochoa and E. M. Atkins, "Urban Metric Maps for Small Unmanned Aircraft Systems Motion Planning," *J. Aerosp. Inf. Syst.*, vol. 19, no. 1, pp. 37–52, Sep. 2021, doi: 10.2514/1.I010979.

[19] C. A. Ippolito, "Dynamic Ground Risk Mitigation for Autonomous Small UAS in Urban Environments," in *AIAA Scitech 2019 Forum*, American Institute of Aeronautics and Astronautics, 2019.

[20] C. Duan, J. Feng, and H. Chang, "Meteorology-Aware Path Planning for the UAV Based on the Improved Intelligent Water Drops Algorithm," *IEEE Access*, vol. 9, pp. 49844–49856, 2021, doi: 10.1109/ACCESS.2021.3068972.

[21] W. H. Al-Sabban, L. F. Gonzalez, and R. N. Smith, "Wind-energy based path planning for Unmanned Aerial Vehicles using Markov Decision Processes," in *2013 IEEE International Conference on Robotics and Automation*, 2013, pp. 784–789, doi: 10.1109/ICRA.2013.6630662.

[22] M. Xue and M. Wei, "Small UAV Flight Planning in Urban Environments," in *AIAA AVIATION 2020 FORUM*, American Institute of Aeronautics and Astronautics, 2020.

[23] D. Hong, S. Lee, Y. H. Cho, D. Baek, J. Kim, and N. Chang, "Energy-Efficient Online Path Planning of Multiple Drones Using Reinforcement Learning," *IEEE Trans. Veh. Technol.*, vol. 70, no. 10, pp. 9725–9740, 2021, doi: 10.1109/TVT.2021.3102589.

[24] J. Lou, B. Yuksek, G. Inalhan, and A. Tsourdos, "An RRT* Based Method for Dynamic Mission Balancing for Urban Air Mobility Under Uncertain Operational Conditions," in *2021 IEEE/AIAA 40th Digital Avionics Systems Conference (DASC)*, 2021, pp. 1–10, doi: 10.1109/DASC52595.2021.9594424.

[25] L. Wang, P. D. Groves, and M. K. Ziebart, "GNSS Shadow Matching: Improving Urban Positioning Accuracy Using a 3D City Model with Optimized Visibility Scoring Scheme," *NAVIGATION*, vol. 60, no. 3, pp. 195–207, Sep. 2013, doi: 10.1002/navi.38.

[26] H. Ko, B. Kim, and S. Kong, "GNSS Multipath-Resistant Cooperative Navigation in Urban Vehicular Networks," *IEEE Trans. Veh. Technol.*, vol. 64, no. 12, pp. 5450–5463, 2015, doi: 10.1109/TVT.2015.2481509.

[27] Y. Zhang, "LILO: A novel Lidar-IMU SLAM System with Loop Optimization," *IEEE Trans. Aerosp. Electron. Syst.*, p. 1, 2021, doi: 10.1109/TAES.2021.3135234.

[28] S. Weiss, D. Scaramuzza, and R. Siegwart, "Monocular-SLAM–based navigation for autonomous micro helicopters in GPS-denied environments," *J. F. Robot.*, vol. 28, no. 6, pp. 854–874, Nov. 2011, doi: 10.1002/rob.20412.

[29] V. O. Sivaneri and J. N. Gross, "UGV-to-UAV cooperative ranging for robust navigation in GNSS-challenged environments," *Aerosp. Sci. Technol.*, vol. 71, pp. 245–255, 2017, doi: 10.1016/j.ast.2017.09.024.

[30] F. Causa and G. Fasano, "Improving Navigation in GNSS-Challenging Environments: Multi-UAS Cooperation and Generalized Dilution of Precision," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 57, no. 3, pp. 1462–1479, 2021, doi: 10.1109/TAES.2020.3043543.

[31] J. Gutierrez, N. A. Neogi, D. Kaeli, and E. T. Dill, "A High-Performance Computing GNSS-aware Path Planning Algorithm for Safe Urban Flight Operations," in *AIAA AVIATION 2022 Forum*, American Institute of Aeronautics and Astronautics, 2022.

[32] Y. Watanabe, A. Veillard, and C. Chanel, "Navigation and Guidance Strategy Planning for UAV Urban Operation," in *AIAA Infotech @ Aerospace*, American Institute of Aeronautics and Astronautics, 2016.

[33] J.-A. Delamer, Y. Watanabe, and C. P. C. Chanel, "Safe path planning for UAV urban operation under GNSS signal occlusion risk," *Rob. Auton. Syst.*, vol. 142, p. 103800, 2021, doi: https://doi.org/10.1016/j.robot.2021.103800.

[34] F. Causa, A. Franzone, and G. Fasano, "Strategic and Tactical Path Planning for Urban Air Mobility: Overview and Application to Real-World Use Cases," *Drones*, vol. 7, no. 1. 2023, doi: 10.3390/drones7010011.

[35] Z. Yao *et al.*, "3DCityDB - a 3D geodatabase solution for the management, analysis, and visualization of semantic 3D city models based on CityGML," *Open Geospatial Data, Softw. Stand.*, vol. 3, no. 1, p. 5, 2018, doi: 10.1186/s40965-018-0046-7.

[36] F. Causa and G. Fasano, "Safe planning and deconfliction for multiple UAVs in high density low altitude urban environments," in *40th Digital Avionics Systems Conference (DASC)*, San Antonio, Texas, 2021.

[37] I. A. Sucan, M. Moll, and L. E. Kavraki, "The Open Motion Planning Library," *IEEE Robot. Autom. Mag.*, vol. 19, no. 4, pp. 72–82, 2012, doi: 10.1109/MRA.2012.2205651.

[38] J. D. Gammell, S. S. Srinivasa, and T. D. Barfoot, "Batch Informed Trees (BIT*): Sampling-based optimal planning via the heuristically guided search of implicit random geometric graphs," in *2015 IEEE International Conference on Robotics and Automation (ICRA)*, 2015, pp. 3067–3074, doi: 10.1109/ICRA.2015.7139620.

[39] W. Chi and M. Q.-H. Meng, "Risk-RRT: A robot motion planning algorithm for the human robot coexisting environment," in *2017 18th International Conference on Advanced Robotics (ICAR)*, 2017, pp. 583–588, doi: 10.1109/ICAR.2017.8023670.

[40] K. Naderi, J. Rajamäki, and P. Hämäläinen, "RT-RRT*: A Real-Time Path Planning Algorithm Based on RRT*," in *Proceedings of the 8th ACM SIGGRAPH Conference on Motion in Games*, 2015, pp. 113–118, doi: 10.1145/2822013.2822036.

[41] D. Shi, X. Dai, X. Zhang, and Q. Quan, "A Practical Performance Evaluation Method for Electric Multicopters," *IEEE/ASME Trans. Mechatronics*, vol. 22, no. 3, pp. 1337–1348, 2017, doi: 10.1109/TMECH.2017.2675913.

[42] A. Godbole, K. Subbarao, A. Dogan, and B. Huff, "Semi-analytical range and endurance computation of battery-powered multi-copter unmanned aerial systems under steady wind conditions," *Proc. Inst. Mech. Eng. Part G J. Aerosp. Eng.*, vol. 233, no. 14, pp. 5282–5294, Apr. 2019, doi: 10.1177/0954410019842714.

[43] J. Gordon Leishman, *Principles of helicopter aerodynamics*, Second edi. New York: Cambridge University Press, 2006.

[44] T. Krebs, G. Bramesfeld, and J. Cole, "Transient Thrust Analysis of Rigid Rotors in Forward Flight," *Aerospace* , vol. 9, no. 1. 2022, doi: 10.3390/aerospace9010028.

[45] Z. Liu, R. Sengupta, and A. Kurzhanskiy, "A power consumption model for multi-rotor small unmanned aircraft systems," in *2017 International Conference on Unmanned Aircraft Systems (ICUAS)*, 2017, pp. 310–315, doi: 10.1109/ICUAS.2017.7991310.

[46] K. Meier, R. Hann, J. Skaloud, and A. Garreau, "Wind Estimation with Multirotor UAVs," *Atmosphere*, vol. 13, no. 4. 2022, doi: 10.3390/atmos13040551.

[47] S. Karaman, M. R. Walter, A. Perez, E. Frazzoli, and S. Teller, "Anytime Motion Planning using the RRT*," in *2011 IEEE International Conference on Robotics and Automation*, 2011, pp. 1478–1483, doi: 10.1109/ICRA.2011.5980479.

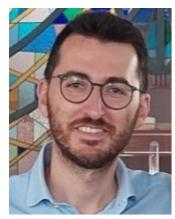
[48] J. D. Gammell, S. S. Srinivasa, and T. D. Barfoot, "Informed RRT*: Optimal sampling-based path planning focused via direct sampling of an admissible ellipsoidal heuristic," in *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2014, pp. 2997–3004, doi: 10.1109/IROS.2014.6942976.

[49] H. Lee, C. M. Harris, A. P. Payan, and D. Mavris, "Risk-Aware Trajectory Planning Using Energy-based Analysis for Aerial Vehicles," in *AIAA AVIATION 2021 FORUM*, American Institute of Aeronautics and Astronautics, 2021.

[50] C. Richter, A. Bry, and N. Roy, "Polynomial Trajectory Planning for Aggressive Quadrotor Flight in Dense Indoor Environments," in *Robotics Research: The 16th International Symposium ISRR*, 2016, pp. 649–666, doi: 10.1007/978-3-319-28872-7_37.

[51] Honeywell, "HG1120 MEMS Inertial Measurement Unit," 2018. https://aerospace.honeywell.com/en/~/media/aerospace/files/brochures/n61-1524-000-004-hg1120-mems-inertial-measurement-unit-bro.pdf (accessed Jun. 10, 2019).

FLAVIA CAUSA is Member of the IEEE AESS Avionics Systems Panel. She received the M.S. degrees in aerospace engineering and the Ph.D. degree in Industrial Engineering from the University of Naples Federico II in 2015 and in 2020. Since 2020, she is a research fellow at University of Naples "Federico II". Her research interests include cooperative navigation of aerospace platforms and GNC for unmanned aerial vehicle. In this framework her research activity is mainly focused on path planning and navigation for UAVs and UAM related topics, such as UAV integration in urban cities. She also deals with problems such as sense and avoid and ground radar surveillance. Dr. Causa was awarded with the Amelia Earhart fellowship in 2019.



GIANCARMINE FASANO (Senior Member IEEE) received the M.S. and Ph.D. degrees in Aerospace Engineering in 2004 and 2007, respectively. He is currently an Associate Professor at the University of Naples "Federico II", where he holds the courses "Unmanned Aircraft Systems" and "Space Flight Dynamics". His research activities in the field of aeronautics are focused on UAS, and in particular on Sense and Avoid and cooperative Multi-UAV Systems. In the space field he mainly works on distributed space systems and proximity operations, with emphasis on relative motion design and control. He is Member of the IEEE AESS Avionics Systems Panel within which he chairs the UAV Committee, Associate Editor of the IEEE AESS Magazine for the UAS area of specialty, IEEE Senior Member. He has been tutorial instructor for the IEEE/AIAA Digital Avionics Systems Conference, and Guest Editor of a Special Issue of the IEEE AESS Magazine dedicated to Sense and Avoid. He is also Member of the AIAA Sensor Systems and Information Fusion Technical Committee, and he has been Member of the IAA Committee on Small Satellites. He coauthored about 150 journal/conference papers and book chapters which gained several citations within the scientific literature, and is a reviewer for top-rated journals in the field of aerospace engineering and robotics.