POLITECNICO DI TORINO
Repository ISTITUZIONALE

Nonvisible Satellite Estimation Algorithm for Improved UAV Navigation in Mountainous Regions

Original
Nonvisible Satellite Estimation Algorithm for Improved UAV Navigation in Mountainous Regions / DE VIVO, F., Battipede, M., Gili, P.. - In: IEEE AEROSPACE AND ELECTRONIC SYSTEMS MAGAZINE. - ISSN 0885-8985. - STAMPA. - 33:11(2018), pp. 4-19. [10.1109/MAES.2018.170220]

Availability:
This version is available at: 11583/2715515 since: 2019-01-09T18:10:35Z

Publisher:
IEEE

Published
DOI:10.1109/MAES.2018.170220

Terms of use:

This article is made available under terms and conditions as specified in the corresponding bibliographic description in the repository

Publisher copyright
IEEE postprint/Author's Accepted Manuscript

©2018 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collecting works, for resale or lists, or reuse of any copyrighted component of this work in other works.

(Article begins on next page)

07 November 2026

Non-visible satellite estimation algorithm for improved UAV navigation in mountainous regions

Francesco De Vivo, Manuela Battipede, Piero Gili

***Abstract*—This paper presents a very simple and computationally efficient algorithm for the calculation of the occlusion points of a scene, observed from a given point of view. This algorithm is used to calculate, in any point of a control volume, the number of visible satellites and the Dilution Of Precision (DOP). Knowledge of these information is extremely important to reject measurements of non-visible satellites and for the reconstruction of a fictitious Digital Elevation Map (DEM), that envelops all the regions characterized by a number of visible satellites lower than a given threshold. This DEM evolves in time according to the platform motion and satellite dynamics. Because of this time dependency, the Digital Morphing Map (DMM) has been defined. When the DMM is available, it can be used by the path planning algorithm to optimise the platform trajectory in order to avoid regions where the number of visible satellites is dramatically reduced, the DOP value is very high and the risk to receive corrupted measurement is large. In this paper also presents the concept of a Safety Bubble Obstacle Avoidance (SBOA) system. This technique takes advantage from the numerical properties of the covariance matrix defined in the Kalman filtering process. A space and time safety bubble is defined according to the DOP value and is used to automatically determine a minimum fly distance from the surrounding obstacles.**

***Index Terms*—Occlusion points, satellite, DOP, DEM, GNSS-denied region, UAV**

I. Introduction

GLOBAL Navigation Satellite System (GNSS) is one of the most used technology around the world for navigation. This technology is integrated in aircraft, ship and submarine guidance systems; it is available in cell phones and cars, or also used to monitor Earth surface movements. The system accuracy spans from a range of a few centimetres up to meters. This variation is related to different sources of error such as the receiver clock error, Earth's ionospheric and tropospheric signal delay [1] or multipath effect in semi-obstructed area. If the surrounding objects have highly reflective surfaces, the error magnitude might surge to several hundred meters due to multiple reflections that the signal experiences before reaching the receiving antenna. A further source of error is present when the same signal is received more than once. As the present work is focused only on the effect of non-visible satellites, multipath effect is not further investigated in this paper. In order to reduce errors related to non-direct signal reception, a proposed solution considers the exclusion of satellites that are not in line of sight (NLOS) with the receiver and the use of the data related to the excluded satellites for state estimation purposes. This data is also used for different scopes, such as creating a time and space dependent safety bubble to improve obstacles safety clearance. Ignoring NLOS satellites can lead to the improvement of the positioning accuracy [2]. However, excluding satellites, the DOP, associated to a bad geometrical repartition of the remaining satellites, could increase, providing a less accurate position estimation. In addition, if less than four satellites are accessible for the localization the service becomes unavailable. In [3], NLOS satellites are detected by projecting the satellite position on a virtual view from the vehicle position. The surrounding features act as a mask. If satellite projections are occluded by the mask, they are excluded. In [4], digital elevation maps and 3D data of the environment from cartography are used in order to identify NLOS signals by ray-tracing technique [5]. In this case, positioning of the receiver in the 3D model is needed. In [6], the 3D digital elevation model is not recorded. In this case, the DEM is built with LIDAR (Light Detection and Ranging) data. The scanned model is used as a mask to determine NLOS satellites. For fixed positions, excluding NLOS signals, an accuracy of less than $1\,\mathrm{m}$ is achieved. In [7], two methods to identify NLOS satellites are compared: the first one is a vision-based method, where the visible structures in roof-mounted fisheye images are processed [8], whereas the second one is based on a Carrier-to-receiver Noise density threshold (CN0) technique. Better results have been achieved with the second method, namely by excluding NLOS signals with the CN0 threshold. In [9] a technique for high-accuracy localization of moving vehicles that utilizes maps of urban environments is proposed. Papers [10] and [11] introduce a novel solution for autonomous navigation of a micro helicopter, through a completely unknown environment, addressed by using a single camera and inertial onboard sensors. The use of a monocular simultaneous localization and mapping (SLAM) framework, to stabilize the vehicle in six degrees of freedom, enables to overcome the problem of platform drift caused by a GNSS loss. A similar approach, based on SLAM is presented in [12] for indoor environments and urban canyon navigation. In [13] and [14] a Visual-Inertial (V-INSs) system is used to sustain prolonged real-world GNSS-denied flights, both for indoor and outdoor GNSS-degraded environments. A different approach, based on stereo camera and laser rangefinder, is adopted in [15], to navigate a quadrotor helicopter in an unstructured and unknown GNSS-denied indoor environment. In [16] a

F. De Vivo is with the Department of Mechanical and Aerospace Engineering, Polytechnic of Turin, Corso Duca degli Abruzzi, 24 - 10129 Torino, Italy e-mail: francesco.devivo@polito.it

M. Battipede is with the Department of Mechanical and Aerospace Engineering, Polytechnic of Turin, Corso Duca degli Abruzzi, 24 - 10129 Torino, Italy e-mail: manuela.battipede@polito.it

P. Gili is with the Department of Mechanical and Aerospace Engineering, Polytechnic of Turin, Corso Duca degli Abruzzi, 24 - 10129 Torino, Italy e-mail: piero.gili@polito.it

novel sensor grid, using ultrasonic transmitters, is presented for GNSS-denied navigation and 3D positioning applications. In [17] a novel information fusion algorithm based on a particle filter (PF) is proposed to achieve lane level tracking accuracy under a GNSS-denied environment. [18] proposes a laser-based tracking system of moving objects using multiple mobile robots as sensor nodes in Global navigation satellite system (GNSS)-denied environments. In [19], the problem of cooperative multi-robot planning in unknown environments is investigated. This is done by actively developing belief spaces planning approaches that account for the different sources of uncertainty within planning and environment. In [20] and [21] a 3D Lidar SLAM system is integrated with GNSS/INS by adapting the measurement noise covariance matrix of an Extended Kalman Filter, to improve accuracy and reliability of the position estimation in urban canyons. Less literature is dedicated to the optimisation of the platform trajectory, starting from the knowledge of the regions of degraded GNSS signal spaces. Gandor in [22] challenged the problem of GNSS signal masking in complex mountainous terrain. Zimmermann et alii [23] presented straightforward approaches, to improve the direct positioning accuracies for UAV-based mapping and surveying applications, under challenging GNSS measurement conditions. Starting from a 3D model of the surrounding buildings and vegetation in the area of interest, a GNSS geometry map is determined and integrated in the flight planning process, to reduce GNSS challenging environments. In [24] the authors created a positioning error distribution, by simulating the multipath effect in a control volume. This error distribution, along with a 3D building model was used for a path planning which avoids buildings and regions with high multipath effects. In [25] a virtual image processing algorithm is used to detect and eliminate possible faulty measurements and NLOS satellites, to improve GNSS navigation in urban canyons. Groves in [26] introduced the concept of intelligent urban positioning (IUP), which combines multi-constellation GNSS, multiple techniques for detecting non-line-of-sight (NLOS) signal propagation and multiple techniques using three-dimensional mapping. In [27] and [28] a number of advances in the shadow-matching algorithm were presented. Furthermore, a new scoring scheme was developed to account for signal diffraction and reflection. In [29] and [30] an efficient smartphone-based shadow-matching positioning system was designed.

Part of the previous works is focused on improving the GNSS accuracy to enhance the local platform position estimation. The remaining part is focused on mapping the GNNS accuracy over a wider area for path planning purposes. The present paper uses both approaches. In particular this paper presents an efficient algorithm to determine the occluded regions for each satellite, starting from the knowledge of a mountain digital elevation model [31], [32]. Based on this occlusion map, a time dependent DEM is designed for obstacle avoidance. Defining for each satellite the invisible regions in a 3D control volume, the DOP value, restricted to the visible satellites, can be calculated. This permits to obtain a 3D map of flight and not-flight zones, according to the local value of the DOP. Once the DOP map is available in the control volume, the real DEM is updated, by considering a surface that interpolates the DOP values that exceed a given threshold. This new time-dependent map is called Digital Morphing Map (DMM). The DMM can be used to optimise aircraft or helicopter trajectories when they are involved in particular risky missions, such as search and rescue operations in mountainous regions, flight support during wildfires [33], [34], or UAVs applications in urban environments. A further advantage is the compatibility of this approach with all the previously described techniques for positioning error reduction. The integration of the proposed algorithm with these techniques has the potential to improve the trajectory planning and, at the same time, to guarantee a higher position accuracy. By calculating the DOP map in the control volume, an obstacle avoidance technique, based on a safety bubble (SBOA), could be developed. The mathematical background of this technique is introduced in this paper. The algorithm, with all the numerical and experimental results, is extensively presented in the following sections.

II. Methodology

This section presents the procedure to calculate and update the satellite positions, starting from the knowledge of the almanac data. It also introduces a method to select the satellites that are potentially visible to a user, located in a specific place on Earth. After this, the algorithm to calculate occlusion points is explained in details. In conclusion, the methodology to calculate the DMM is presented.

A. Visible GNSS calculation

In order to calculate and update the satellite position, the orbital parameters, listed in TABLE I, are required. These values can be downloaded from the CelesTrack website [35].

TABLE I
SEM ALMANAC [35]

SYMBOL	DEFINITION	UNIT
N_{SAT}	Number of satellites	–
WN	Week number	–
t_{oa}	Time of applicability	sec
PRN	Pseudo-random number	–
SVN	SVN number	–
URA	URA number	–
e	Orbital eccentricity	–
i_0	Inclination offset	sc
$\dot{\Omega}$	Rate of right ascension	sc/sec
$\sqrt{a}$	Square-root of semimajor axis	m
Ω_0	Longitude of orbital plane	sc
ω	Argument of perigee	sc
M_0	Mean anomaly	sc
a_{f_0}	Zero-order clock correction	sec
a_{f_1}	First-order clock correction	sec/sec
L	Satellite health	–
SC	Satellite configuration	–

The almanac is used to simulate the entire constellation. In case of a real application, where the more accurate ephemeris data, sent by the satellites, are available, their use is mandatory. For simulation purposes, the use of a set of data with respect to the other is irrelevant, preserving the solution consistency.

The complete procedure to estimate the satellite position in the Earth-Centered Earth-Fixed (ECEF) reference frame, starting from the knowledge of the almanac data, is extensively described in [36] and reported for sake of clarity in TABLE II.

TABLE II
GNSS POSITION CALCULATION [36]

$a = (\sqrt{a})^2$	Semi-major axis
$n_0 = \sqrt{\frac{\mu}{a^3}}$	Mean motion
$\delta t = t - t_{oe}$	Time from Ephemeris time (ET)
$M = M_0 + n_0 \delta t$	Mean anomaly
$M = E - e \sin E$	Kepler's equation
$\nu = \tan^{-1}\left(\frac{\sin E \sqrt{1-e^2}}{\cos E - e}\right)$	True anomaly
$\Phi = \nu + \omega$	Argument of latitude
$r = a(1 - e \cos E)$	Earth-Sat radius
$x_o = r \cos \Phi$ $y_o = r \sin \Phi$	Orbital plane coord.
$\Omega = \Omega_0 + (\dot{\Omega} - \dot{\Omega}_e)\delta t - \dot{\Omega}_e t_{oe}$	Longitude of Ascending Node
$\begin{bmatrix} x^E \\ y^E \\ z^E \end{bmatrix} = \begin{bmatrix} c_\Omega & -c_{i_0} s_\Omega \\ s_\Omega & c_{i_0} c_\Omega \\ 0 & s_{i_0} \end{bmatrix} \begin{bmatrix} x_o \\ y_o \end{bmatrix}$	ECEF satellite coordinates

Once the position of every satellite is obtained in the ECEF reference frame, the satellites that are visible from a point $\mathbf{P}$ on the Earth surface can be calculated.

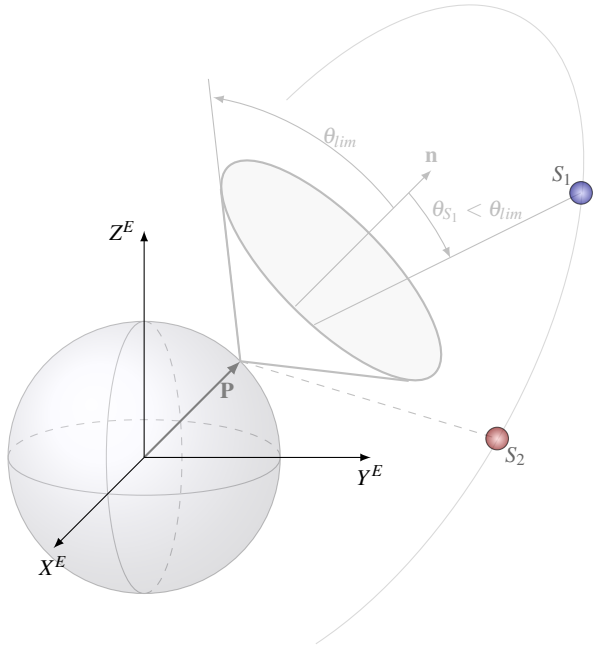


Fig. 1. Visible satellites calculation using the visibility cone. Visible satellites are those with $\theta_{S_i} < \theta_{lim}$

These satellites are those with an elevation angle above the horizon in the range 5 deg and 25 deg [37]. These values are determined by the link budget and terrain orography. Once the elevation angle is known, a visibility cone can be defined. The cone angular aperture is θ_{lim} and is defined with respect to the vector direction $\mathbf{n}$ as shown in Fig. 1. Every satellite inside this cone is potentially visible from a receiver in $\mathbf{P}$ (such as S_1 in Fig. 1), whereas those that orbit outside of it are not visible (such as S_2 in Fig. 1). In order to determine if a satellite is visible or not, the following condition has to be verified

$$\arccos\left(\mathbf{n} \cdot \frac{\mathbf{r}_{S_i} - \mathbf{P}}{\|\mathbf{r}_{S_i} - \mathbf{P}\|}\right) = \theta_{S_i} < \theta_{lim} \tag{1}$$

where $\mathbf{n} = \mathbf{P}/\|\mathbf{P}\|$ and the vector $\mathbf{r}_{S_i} = \begin{bmatrix} x^E_{S_i} & y^E_{S_i} & z^E_{S_i} \end{bmatrix}$ contains the ECEF coordinates of the satellite S_i. The point $\mathbf{P}$ is the position on the Earth surface of the GNSS receiver. In a more general situation, the vector $\mathbf{n}$ is the pointing direction of the GNSS receiver antenna.

B. Occlusion-points determination algorithm

In this section, the algorithm developed for the calculation of the occluded points is presented. This algorithm has a computational complexity of $\mathcal{O}(N)$.
The first step of the algorithm is to make the process column independent. This step allows the parallel processing of each column of the DEM matrix. This is achieved by resampling the DEM on a non-Cartesian grid, whose cells are the projection of the image plane on ground. A simplification of this 3D grid is represented in yellow in Fig. 2. A local reference frame $x^f - y^f$ is defined for each slice on ground, relative to the projection of a single matrix column. The origin of this reference frame is the nearest grid cell to the projection of the satellite S_i on ground. x^f is parallel to the main slice axis, whereas y^f is orthogonal to it. The algorithm processes each column starting from the lowest absolute values of x^f, which means that from the example presented in Fig. 2, the points are processed from P_1 to P_6. For the three points considered in Fig. 2, P_6 is occluded, whereas P_1 and P_5 are visible. This effect is relative to a vertical switch in the point position on the image plane $x^c - y^c$. Indeed, $x^f_{P_6} > x^f_{P_5}$, whereas $x^c_{P_6} < x^c_{P_5}$. The point coordinates in the camera reference frame (CRF) are obtained by means of a homogeneous transformation f

$$\begin{bmatrix} x^c \\ y^c \\ z^c \\ 1 \end{bmatrix} = f(\mathbf{P}) = \mathbf{T}^c_I \begin{bmatrix} \mathbf{P} \\ 1 \end{bmatrix} = \begin{bmatrix} \mathbf{R}^c_I & -\mathbf{d}^c_I \\ \mathbf{0}_{1x3} & 1 \end{bmatrix} \begin{bmatrix} x^I \\ y^I \\ z^I \\ 1 \end{bmatrix}, \tag{2}$$

where $\mathbf{R}^c_I$ and $\mathbf{d}^c_I$ are respectively the rotational matrix and the translational vector (resolved in the camera reference frame) which take into account the different orientation and origins of the two reference frames. $\mathbf{P}$ is the vector containing the coordinates of the generic point P_{ij} of the DEM in the IRF. The IRF is the reference frame used to represent the DEM. It could be either a Geodetic or a North-East-Down (NED) or ECEF reference frame. The homogeneous transformation $\mathbf{T}^c_I$ is given by

$$\mathbf{T}^c_I = \mathbf{T}^c_g \mathbf{T}^g_b \mathbf{T}^b_I = \begin{bmatrix} \mathbf{R}^c_g & -\mathbf{d}^c_g \\ \mathbf{0}_{1x3} & 1 \end{bmatrix} \begin{bmatrix} \mathbf{R}^g_b & -\mathbf{d}^g_b \\ \mathbf{0}_{1x3} & 1 \end{bmatrix} \begin{bmatrix} \mathbf{R}^b_I & -\mathbf{d}^b_I \\ \mathbf{0}_{1x3} & 1 \end{bmatrix} \tag{3}$$

where $\mathbf{T}^b_I$ projects the points from the inertial to satellite (body) reference frame, $\mathbf{T}^g_b$ from body to gimbal and $\mathbf{T}^c_g$ from

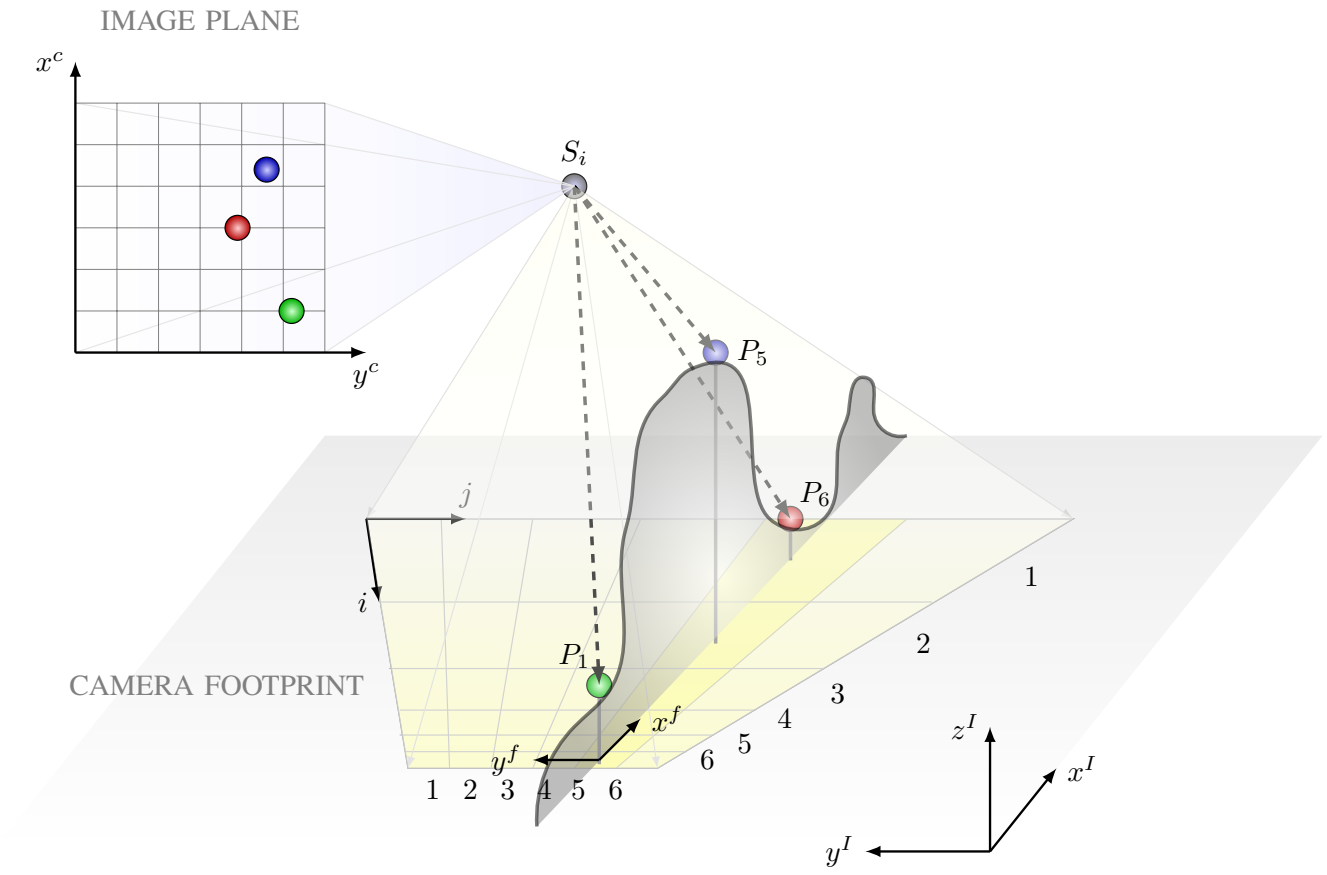


Fig. 2. Mesh geometry based on camera footprint and image formation by means of homogeneous transformation f. In the figure there are also the image place $x^c - y^c$, the DEM matrix with the coordinate in the IRF ($\mathbf{X^I}_{DEM}$) and CRF ($\mathbf{X^C}_{DEM}$)

gimbal to camera. The rotation matrix $\mathbf{R}_I^b$ is given by the sequence of Euler rotation 313

$$\mathbf{R}_I^b = \begin{bmatrix} c_\phi c_\psi - s_\phi c_\theta s_\psi & s_\phi c_\theta c_\psi + c_\phi s_\psi & s_\phi s_\theta \\ -c_\phi c_\theta s_\psi - s_\phi c_\psi & c_\phi c_\theta c_\psi - s_\phi s_\psi & c_\phi s_\theta \\ s_\theta s_\psi & -s_\theta c_\psi & c_\theta \end{bmatrix}, \quad (4)$$

the matrix $\mathbf{R}_b^g$ is

$$\mathbf{R}_b^g = \begin{bmatrix} c_{\theta_{el}} c_{\psi_{az}} & c_{\theta_{el}} s_{\psi_{az}} & s_{\theta_{el}} \\ -s_{\theta_{el}} & c_{\psi_{az}} & 0 \\ -s_{\theta_{el}} c_{\psi_{az}} & -s_{\theta_{el}} s_{\psi_{az}} & c_{\theta_{el}} \end{bmatrix} \quad (5)$$

where θ_{el} is the gimbal elevation angle, defined in the $x^b - z^b$ plane, and ψ_{az} is the gimbal azimuth angle defined in the $x^b - y^b$ plane. Considering that the satellites are generally off-nadir with respect to the UAV platform, their FOV needs to be centred around $\mathbf{P}$. This is achieved by introducing a gimbal rotation that aligns the camera principal axis with the vector between the camera and the point $\mathbf{P}$. This rotation justifies the use of $\mathbf{T}_b^g$ in Eq. 3. The angles ψ_{az} and θ_{el} used in Eq. 5 are calculated by simple geometric relations as follows

$$\begin{aligned} \psi_{az} &= \arctan 2 \left(\frac{y_{S_i} - y_P}{x_{S_i} - x_P} \right) \\ \theta_{el} &= \arccos \left(\frac{z_P - z_{S_i}}{\|\mathbf{r}_{S_i} - \mathbf{P}\|} \right) - \frac{\pi}{2} \end{aligned} \quad (6)$$

where the $\arctan 2$ indicates the 4-quadrant inverse tangent. Considering that Eq. 6 is applied only to the satellites in the visibility cone, the maximum value of ψ_{az} is bounded by the visibility cone aperture shown in Fig. 1. This is true if the main lobe of the satellite antenna points toward the Earth centre. The matrix $\mathbf{R}_g^c$ is constant if the relative motion between the antenna and gimbals is neglected

$$\mathbf{R}_g^c = \begin{bmatrix} 0 & 0 & -1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{bmatrix}. \quad (7)$$

In the case the DEM is defined in a reference frame different from ECEF, a further transformation is required. The translational vectors $\mathbf{d}_I^b$ in Eq. 3 for a generic satellite is

$$\mathbf{d}_I^b = \mathbf{R}_I^b \mathbf{r}_{S_i}. \quad (8)$$

The same concept applies to $\mathbf{d}_b^g$ and $\mathbf{d}_g^c$. The last parameter to be determined in order to completely characterize the camera model is the Field Of View (FOV). This value controls the extension of the DEM surface to be processed. The goal is to solve the occlusion problem in a control volume $\mathbf{\Omega}$. To keep only $\mathbf{\Omega}$ in the FOV, some control parameters have to be chosen. Considering that the algorithm is built on the camera model, the FOV value is dependent on the focal length, size and number of pixels. Keeping fixed the size and number of

pixels, the only free variable to define the FOV is the focal length. This value is calculated from similarity triangles, by assuming that the ratio between the focal length and the image plane size is proportional to the ratio between the distance of the satellite from the point $\mathbf{P}$ and the control volume width $\delta_{\mathbf{\Omega}}$. By applying this rule, an approximative value of the focal length f_{c_i} for $i = 1 \dots N_s$, is calculated as follows

$$f_{c_i} \approx \frac{n_p \delta_p \|\mathbf{r}_{S_i} - \mathbf{P}\|}{\delta_{\mathbf{\Omega}}}, \tag{9}$$

where n_p is the maximum number of pixels between n_{p_x} and n_{p_y} and δ_p is the pixel dimension. Once the x^c coordinates of the projected points are available, it is possible to discriminate between visible and occluded points. Moving along x^f, as already mentioned, the x^c values of two consecutive points are compared; if the value of x^c relative to the second grid point is lower that the value of x^c relative to the first one, the second point is visible and its x^c value is stored in a temporary variable x^c_{max} and used for the next check. In the case the value of x^c relative to the second point is lower than the first one, the point is occluded and x^c_{max} remains unchanged. From a mathematical point of view

$$x^c_{max} = \max\left(x^c_{P_{k+1}}, x^c_{max}\right) = \max\left(f\left(\mathbf{P}_{k+1}\right) \cdot \mathbf{e}_1, x^c_{max}\right) \tag{10}$$

$$\mathbf{P}^{occ}_{k+1} = \begin{cases} 0 & x^c_{max} = x^c_{P_{k+1}} \\ 1 & \text{otherwise} \end{cases} \tag{11}$$

where $\mathbf{e_1} = [1, 0, 0, 0]$ and $\mathbf{P}^{occ}$ is a boolean vector which stores the positions of the occluded points. Going back to the example of Fig. 2, starting from P_1, the value of x^c_{max} is equal to $x^c_{P_1}$, namely $x^c_{max} = x^c_{P_1}$. Moving to the second point, $x^c_{P_5} > x^c_{max}$ (blue point above green point), then P_5 is visible and $x^c_{max} = x^c_{P_5}$. Moving forward, it happens that the value of $x^c_{P_6} < x^c_{P_5} = x^c_{max}$, and, as a consequence, P_6 is occluded.

An alternative approach to calculate the occlusion points is to use the ray-triangle intersection algorithm [38]. This algorithm verifies if there is an intersection between a ray $\mathbf{r}(t)$ and a triangle, defined by its three vertices V_0, V_1 and V_2. The ray $\mathbf{r}(t)$ is defined as

$$\mathbf{r}(t) = \mathbf{O} + t\mathbf{D} \tag{12}$$

where the vectors $\mathbf{O}$ and $\mathbf{D}$ are the ray origin and direction. A generic point $\mathbf{P}_T(u, v)$ on a triangle can be defined as a function of its coordinates (u, v) as follows

$$\mathbf{P}_T(u, v) = (1 - u - v)\mathbf{V}_0 + u\mathbf{V}_1 + v\mathbf{V}_2, \tag{13}$$

where $\mathbf{V}_0, \mathbf{V}_1$ and $\mathbf{V}_2$ are the triangle vertex coordinates. Equating Eq. 12 and Eq. 13, the following linear system in the variables t, u, and v is obtained

$$\begin{bmatrix} -\mathbf{D} & \mathbf{V}_1 - \mathbf{V}_0 & \mathbf{V}_2 - \mathbf{V}_0 \end{bmatrix} \begin{bmatrix} t \\ u \\ v \end{bmatrix} = \mathbf{O} - \mathbf{V}_0. \tag{14}$$

The solution of this system provides the coordinates of the intersection points. If there is an occluded point along the ray, more than one intersection is found. In this case, the visible point is the one relative to the lowest value of the variable t, whereas all the others are the occluded points. A drawback of this algorithm is that, given a single ray, the system of Eq. 14 has to be solved for all triangular patches representing the DEM surface. Considering that this process has to be repeated for each ray of the image plane and for each visible satellite, it could be computationally demanding. For this reason the proposed solution has been preferred. A comparison of the two algorithms can be found in [31].

C. Morphing surface determination

The Not-Flight Zones (NFZ) are defined as part of a volume where the DOP of visible satellites is greater than a given threshold. In order to calculate the NFZ, the first step is to define the control volume $\mathbf{\Omega}$. The volume is centred around the receiver, and the three dimensions are chosen according to the receiver dynamics: the slower the motion along a given direction, the smaller the dimension in that direction. This choice is driven by the idea that the control volume is the 3D space, inside of which the trajectory must be locally optimised. Considering that the occluded regions inside the volume are function of the satellites motion, it is useless to increase the computational effort by processing regions that can not be physically reached by the platform, before a consistent variations in the not-flight map is experienced. Once the boundaries of $\mathbf{\Omega}$ have been defined, the occlusion mask $\mathbf{P^{occ}}$ can be calculated for each satellite, by using the algorithm described in the previous section. When the occluded points on the DEM surface are available, it is easy to extend this property to the entire volume (hatched area in Fig. 3). The upper bound of these regions are characterized by the rays r_{SP} connecting the satellite position to the mountain peak that fore-runs the first invisible points, as shown in Fig. 3. In this figure, for each satellite, a different pattern highlights the corresponding invisible regions. In the white area of the control volume, all three satellites are in view. The number of visible satellites is indicated by the circled number in that region. In the overlapped regions, more than one satellite is obscured by the DEM. The worst situation is experienced in the valley between the two mountains, where there are zero satellites in view. In order to determine the *flight zones* (FZ) and *not-flight zones*, the DOP value associated to the visible satellites in each of these regions has to be calculated. The no-flight zones are those having a DOP greater than a given threshold DOP_{lim}

$$\mathbf{P}_{NFZ} = \{\, \mathbf{P} \in \mathbf{\Omega} \mid DOP(\mathbf{P}) \geq DOP_{lim} \,\}. \tag{15}$$

The DOP for a given point $\mathbf{P} = \begin{bmatrix} x & y & z \end{bmatrix}$ is

$$DOP = \sqrt{\sigma_x^2 + \sigma_y^2 + \sigma_z^2} \tag{16}$$

where the σ_i are the diagonal elements of the matrix $\mathbf{Q_s}$

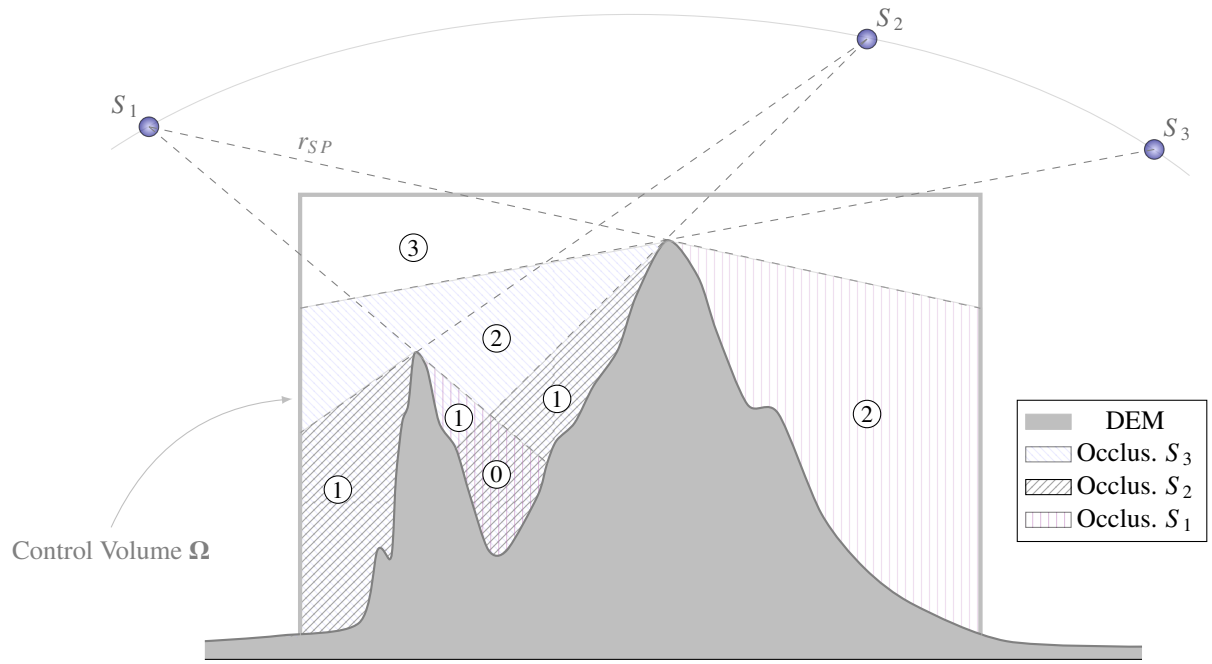


Fig. 3. Occluded regions for three different satellites. The circled numbers indicate the effective number of visible satellites in that region of the control volume $\mathbf{\Omega}$

$$\mathbf{Q_s} = \begin{bmatrix} \sigma_x^2 & \sigma_{xy} & \sigma_{xz} & \sigma_{xt} \\ \sigma_{xy} & \sigma_y^2 & \sigma_{yz} & \sigma_{yt} \\ \sigma_{xz} & \sigma_{yz} & \sigma_z^2 & \sigma_{zt} \\ \sigma_{xt} & \sigma_{yt} & \sigma_{zt} & \sigma_t^2 \end{bmatrix}.$$

This matrix is given by the following relation [39]

$$\mathbf{Q_s} = \left(\mathbf{A}^T\mathbf{A}\right)^{-1} \tag{17}$$

where $\mathbf{A}$ is

$$\mathbf{A} = \begin{bmatrix} \frac{x_{S_1}-x}{\rho_1} & \frac{y_{S_1}-y}{\rho_1} & \frac{z_{S_1}-z}{\rho_1} & -1 \\ \frac{x_{S_2}-x}{\rho_2} & \frac{y_{S_2}-y}{\rho_2} & \frac{z_{S_2}-z}{\rho_2} & -1 \\ & \vdots & & \\ \frac{x_{S_i}-x}{\rho_i} & \frac{y_{S_i}-y}{\rho_i} & \frac{z_{S_i}-z}{\rho_i} & -1 \\ & \vdots & & \\ \frac{x_{S_{N_s}}-x}{\rho_{N_s}} & \frac{y_{S_{N_s}}-y}{\rho_{N_s}} & \frac{z_{S_{N_s}}-z}{\rho_{N_s}} & -1 \end{bmatrix}.$$

The value of ρ_i is given by

$$\rho_{i=1\ldots N_s} = \sqrt{(x_{S_i}-x)^2 + (y_{S_i}-y)^2 + (z_{S_i}-z)^2} \tag{18}$$

where N_s is the number of visible satellites in the point $\mathbf{P}$ of the control volume. The ECEF coordinate superscript E has been omitted for simplicity of notation.

In order to speed-up the code execution, if the control volume dimension is not very large ($\approx 1\,\mathrm{km}$), it is possible to assume that the DOP does not change significantly from point to point, so that its value can be assumed constant over the region, with equal number of visible satellites. This approximation requires, for the example shown in Fig. 3, to calculate 6 values of the DOP (excluding the region with zero visible satellites) instead of calculating it for each cell of the volumetric mesh of $\mathbf{\Omega}$. The validity of this assumption is confirmed by the analysis of the DOP distribution presented in Section IV. The schematic situation presented in Fig. 3 has only the intention to highlight some algorithm peculiarities. The number of satellites has been set equal to 3 to provide the reader with an explicative figure. Once the 3D DOP map inside the control volume is calculated, the DMM can be obtained as the isosurface enveloping the points $\mathbf{P}_{DMM}$, where

$$\mathbf{P}_{DMM} = \{\, \mathbf{P} \in \mathbf{\Omega} \mid DOP(\mathbf{P}) = DOP_{lim} \,\}. \tag{19}$$

This new DEM has been called DMM in Section I, because the surface shape is a time-dependent function of the satellite motion. This technique is very useful to solve the problem related to non direct signal reception. If the platform is flying in the rightmost region of Fig. 3, the signal of the satellite S_1 is shielded by the mountain. In this situation, if the user antenna receives a signal from this satellite, certainly it is corrupted by multiple reflections because the satellite is in NLOS. For this reason, in order to avoid a measure degradation, the NLOS information can be used to filter-out the wrong measurements to mitigate this effect. When the DMM is available, it can be used by the path planning algorithm to optimise the platform trajectory, in order to avoid regions where the number of visible satellites is dramatically reduced, the DOP value is very high and the risk of receiving corrupted measurement is large.

III. Safety Bubble Obstacle Avoidance (SBOA)

In this section an interesting application, deriving from the previous algorithm, is described. This application refers to the

intrinsic capability of an unmanned platform of avoiding an obstacle. To understand how the algorithm works, a platform equipped with a navigation system that integrates measurements from a GNSS and Inertial Measurement Unit (IMU) is considered. Sensor integration is generally performed using a Kalman Filter (KF), or more advanced and sophisticated integration techniques such as the one described in [40], [41]. The filter output is generally characterised by the estimated state vector $\hat{\mathbf{x}}_{n|n}$ and a covariance matrix $\mathbf{P^{xx}}_{n|n}$ that quantifies the error associated to the state vector estimation. The Linear Kalman Filter algorithm is shown in TABLE III.

TABLE III
LKF ALGORITHM [40]

$\hat{\mathbf{x}}_{n\|n-1}$	$= \mathbf{F}\hat{\mathbf{x}}_{n-1\|n-1} + \mathbf{u}_n$
$\mathbf{P^{xx}}_{n\|n-1}$	$= \mathbf{F}\mathbf{P^{xx}}_{n-1\|n-1}\mathbf{F}^T + \mathbf{Q}$
$\hat{\mathbf{z}}_{n\|n-1}$	$= \mathbf{H}\hat{\mathbf{x}}_{n\|n-1}$
$\mathbf{P^{zz}}_{n\|n-1}$	$= \mathbf{H}\mathbf{P^{xx}}_{n\|n-1}\mathbf{H}^T + \mathbf{R}$
$\mathbf{P^{xz}}_{n\|n-1}$	$= \mathbf{P^{xx}}_{n\|n-1}\mathbf{H}^T$
$\mathbf{K_n}$	$= \mathbf{P^{xz}}_{n\|n-1}(\mathbf{P^{zz}}_{n\|n-1})^{-1}$
$\hat{\mathbf{x}}_{n\|n}$	$= \hat{\mathbf{x}}_{n\|n-1} + \mathbf{K_n}(\mathbf{z} - \hat{\mathbf{z}}_{n\|n-1})$
$\mathbf{P^{xx}}_{n\|n}$	$= \mathbf{P^{xx}}_{n\|n-1} - \mathbf{K_n}\mathbf{P^{zz}}_{n\|n-1}\mathbf{K_n}^T$

The matrix $\mathbf{F}$ is the state transition matrix, $\mathbf{u}_n$ is a known control vector, $\mathbf{Q}$ is the noise covariance matrix that takes into account the error associated to the inaccuracy of the dynamic model, $\mathbf{H}$ is the sensor measurement matrix, $\mathbf{R}$ is the sensor noise covariance matrix, $\hat{\mathbf{z}}_{n|n-1}$ is the predicted measurement vector, $\mathbf{z}$ is the vector containing the sensor measurements and $\mathbf{K_n}$ is the Kalman gain. The updated state covariance matrix

$$\mathbf{P^{xx}}_{n|n} = \mathbf{P^{xx}}_{n|n-1} - \mathbf{K_n}\mathbf{P^{zz}}_{n|n-1}\mathbf{K_n}^T \tag{20}$$

is a function of both the Kalman gain $\mathbf{K_n}$, the innovation matrix $\mathbf{P^{zz}}_{n|n-1}$ and the predicted state covariance matrix $\mathbf{P^{xx}}_{n|n-1}$. Substituting in the second term of Eq. 20 the relative expressions of $\mathbf{K_n}$ and $\mathbf{P^{zz}}_{n|n-1}$, taken from TABLE III, the following formulation is obtained

$$\mathbf{K}\mathbf{P^{zz}}\mathbf{K}^T = \left[\mathbf{P^{xz}}(\mathbf{P^{zz}})^{-1}\right]\mathbf{P^{zz}}\left[\mathbf{P^{xz}}(\mathbf{P^{zz}})^{-1}\right]^T = \frac{\mathbf{P^{xx}}\mathbf{H}^T}{\mathbf{H}\mathbf{P^{xx}}\mathbf{H}^T + \mathbf{R}}\left(\mathbf{H}\mathbf{P^{xx}}\mathbf{H}^T + \mathbf{R}\right)\left(\frac{\mathbf{P^{xx}}\mathbf{H}^T}{\mathbf{H}\mathbf{P^{xx}}\mathbf{H}^T + \mathbf{R}}\right)^T. \tag{21}$$

In Eq. 21, the subscript time steps n have been omitted for clarity. Defining

$$\begin{cases}\mathbf{C_1} = \mathbf{P^{xx}}\mathbf{H}^T \\ \mathbf{C_2} = \mathbf{H}\mathbf{P^{xx}}\mathbf{H}^T\end{cases}$$

and substituting them in Eq. 21 the following equation is obtained

$$\mathbf{K}\mathbf{P^{zz}}\mathbf{K}^T = \frac{\mathbf{C_1}}{\mathbf{C_2} + \mathbf{R}}\left(\mathbf{C_2} + \mathbf{R}\right)\left(\frac{\mathbf{C_1}}{\mathbf{C_2} + \mathbf{R}}\right)^T. \tag{22}$$

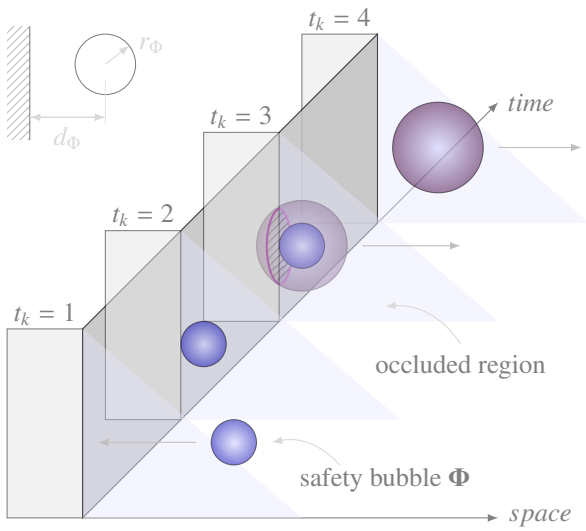


Fig. 4. Safety bubble inflation when the receiver moves in the occluded region

Substituting Eq. 22 in Eq. 20, the state covariance matrix $\mathbf{P^{xx}}_{n|n}$ is written as function of the sensor covariance matrix $\mathbf{R}$ as follows

$$\mathbf{P^{xx}}_{n|n} = \mathscr{F}(\mathbf{R}) = \mathbf{P^{xx}}_{n|n-1} - \frac{\mathbf{C_1}\mathbf{C_1}^T}{(\mathbf{C_2} + \mathbf{R})^T}. \tag{23}$$

As already introduced, the matrix $\mathbf{R}$ quantifies the sensor accuracy. This implies that the higher the sensor measurement error, the higher the matrix values. For simplicity, a scalar problem is considered here, so the matrices of Eq. 23 reduce to scalar values. If two sensors are considered, where the first one is more accurate than the second one, it results that $\mathbf{R_1} < \mathbf{R_2}$, where $\mathbf{R_i}$ (with $i = \{1, 2\}$) indicates the covariance value associated to the sensor i. Substituting these two parameters in Eq. 23 the following inequality is verified

$$\mathbf{P_1^{xx}}_{n|n} < \mathbf{P_2^{xx}}_{n|n}. \tag{24}$$

As the state covariance matrix $\mathbf{P^{xx}}$, associated to the sensor #2, is bigger than the one associated to sensor #1, the state estimation uncertainty associated to sensor #2 is greater than the uncertainty associated to sensor #1. Considering the diagonal elements of $\mathbf{P^{xx}}$, it is possible to draw an n-dimensional ellipsoid that defines the confidence domain around the estimated vector $\hat{\mathbf{x}}$. This ellipsoid, or *safety bubble*, is indicated as $\mathbf{\Phi}(\sigma)$, where $\sigma = \begin{bmatrix}\sigma_{\hat{x}} & \sigma_{\hat{y}} & \sigma_{\hat{z}}\end{bmatrix}$ is the vector of the predicted standard deviations, obtained from the square root of the diagonal elements of $\mathbf{P^{xx}}_{n|n}$. From the inequality relation of Eq. 24 it is easy to figure out that the safety bubble associated to $\mathbf{R_2}$ is bigger than the one associated to $\mathbf{R_1}$.
Starting from these considerations, the safety bubble obstacle avoidance (SBOA) algorithm can be explained. The example shown in Fig. 4 is considered. If the platform flies in normal conditions, with all the satellites in view, the sensor accuracy is the highest, and consequently the safety bubble has the minimum dimensions. For simplicity, we assume that $\sigma_{\hat{x}} = \sigma_{\hat{y}} = \sigma_{\hat{z}}$, for this reason the safety ellipsoid degenerates

into a sphere with radius r_Φ. In order to take into account also the DEM uncertainty (it can span from few centimetres to some metres), the safety bubble is increased accordingly. By including the DEM uncertainty in the safety bubble, the DEM uncertainty becomes zero. An obstacle avoidance algorithm has the aim to optimise the platform trajectory, or the evasive manoeuvre, so that the distance d_Φ between the platform and the obstacle is greater than the safety bubble size

$$d_\Phi \geq r_\Phi. \tag{25}$$

This condition is shown for the time step $t_k = 1$ in Fig. 4. Considering that the safety bubble radius is a confidence interval associated to a variance σ, its value is calculated once a probability $p(\sigma)$ is chosen. For safety reasons, it is not allowed to violate the inequality of Eq. 25, and the minimum flying distance is $d_\Phi = r_\Phi$, shown in Fig. 4 for $t_k = 2$. If the platform moves inside the satellite occluded region, the conventional navigation system do not change the size of the safety bubble associates to the estimated state $\hat{\mathbf{x}}$ ($t_k = 3$) because $\mathbf{R}$ is kept constant. In the real case, the GNSS performance is degraded because of the occlusion effect and the increase of the DOP caused by the reduction of visible satellites. By including this sensor degradation information in the measurement covariance matrix $\mathbf{R}$, the safety bubble increases as shown by the Eq. 24 (violet ball for $t_k = 3$ in Fig. 4). By neglecting this bubble inflation, the navigation system performs a manoeuvre with the constrain $d_\Phi = r_\Phi$, where r_Φ is the value associated to the nominal situation. By performing this manoeuvre, the real safety bubble intersects the obstacle surface. This means that the probability of impacting with the obstacle is higher than the probability assured by the safety bubble radius. If the SBOA algorithm is applied, as soon as that the platform flies in the occluded region, the DOP value in that region increases the covariance matrix $\mathbf{R}$, increasing immediately the safety bubble dimensions. As a consequence of this bubble inflation, the minimum allowed distance d_ϕ becomes bigger than the previous one, pushing the platform far from the obstacle surface ($t_k = 4$). This "repulsive effect" increases with increasing values of the local DOP, forcing the body to fly at a safer distance from the obstacle.

In Fig. 5, the block scheme that summarizes the procedure to calculate the SBOA is presented. The input values are the satellite constellation, DEM and UAV position. The algorithm to calculate the occlusion points, and consequently the non visible satellites in each point of the control volume $\mathbf{\Omega}$, is applied. At this point, a list of visible satellites is associated to each voxel of $\mathbf{\Omega}$. By applying Eq. 16 to the visible satellites in each voxel, the DOP map is obtained. From this map the DMM can be calculated using Eq. 15. The measurement covariance matrix $\mathbf{R}$ is dynamically update according to the local value of the DOP as follows

$$\mathbf{R} = \text{DOP} \begin{bmatrix} \mu_1 & 0 & 0 & \dots & 0 \\ 0 & \mu_2 & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & \mu_{N_s} \end{bmatrix} \tag{26}$$

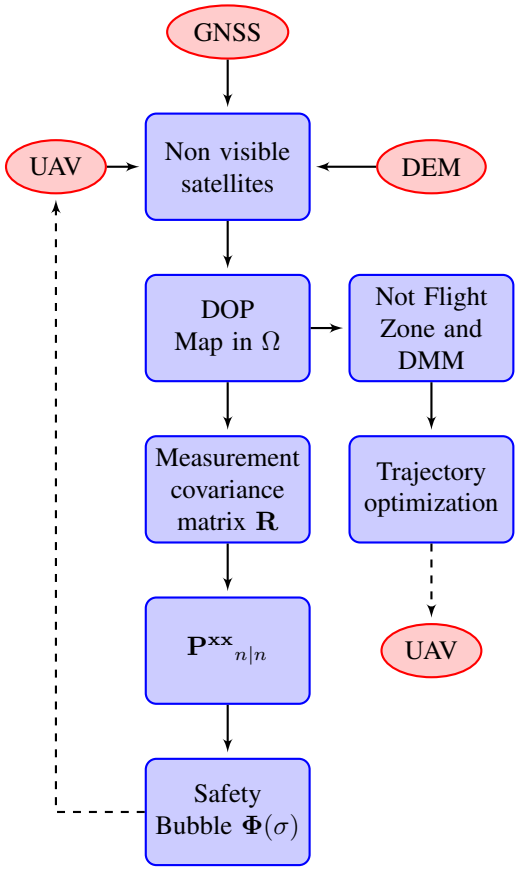


Fig. 5. Scheme to calculate the Safety Bubble and Digital Morphing Model

where μ is a proportionality coefficient. Using this matrix in Eq. 23, the state covariance matrix $\mathbf{P^{xx}}_{n|n}$ can be calculated. The standard deviations $\sigma = \begin{bmatrix} \sigma_{\hat{x}} & \sigma_{\hat{y}} & \sigma_{\hat{z}} \end{bmatrix}$, required for the calculation of the safety bubble $\mathbf{\Phi}(\sigma)$, is obtained by taking the square root of the diagonal components of $\mathbf{P^{xx}}_{n|n}$. A more conservative approach, adopted in this paper, is to calculate a spherical safety bubble, for which the radius is

$$r_\Phi = \epsilon \sqrt{\sum \text{diag}\left(\mathbf{P^{xx}}_{n|n}\right)}. \tag{27}$$

where ϵ is an integer multiplicative coefficient, which is used to regulate the confidence interval.

IV. Numerical Results

This paragraph shows how the algorithm behaves when a more realistic scenario is presented. The 3D Digital Elevation Map is part of the Mount Mitchell in Yancey County (North Carolina), digitalized with a map resolution of $\approx$30 m. For real applications the use of a more accurate DEM is required. This particular scenario has been chosen in order to highlight the potentiality of the algorithm when a very complex scene is processed. The mountainous scenario is preferred due to the variety of missions that can be performed performed, involving vehicles, pedestrians or different kind of aircraft. In order to project the DEM in the CRF, the homogeneous transformation of Eq. 2 has been used. For this simulation, the control volume dimensions equate exactly the map size, imposing the same

mesh discretization. The control volume resolution along the three Cartesian axes in meters is $\delta_{\mathbf{\Omega}} = \begin{bmatrix} 48.5 & 49 & 12 \end{bmatrix}$. The DEM dimensions are respectively $10\,\mathrm{km} \times 13\,\mathrm{km}$, whereas the maximum mountain altitude above the World Geodetic System WGS84 is $\approx$2000 m. The number of pixels along x^c and y^c are respectively $n_{p_x} = n_{p_y} = 300$, whereas the pixel dimensions are $\delta_{p_x} = \delta_{p_y} = 3.6 \times 10^{-6}\,\mathrm{m}$.

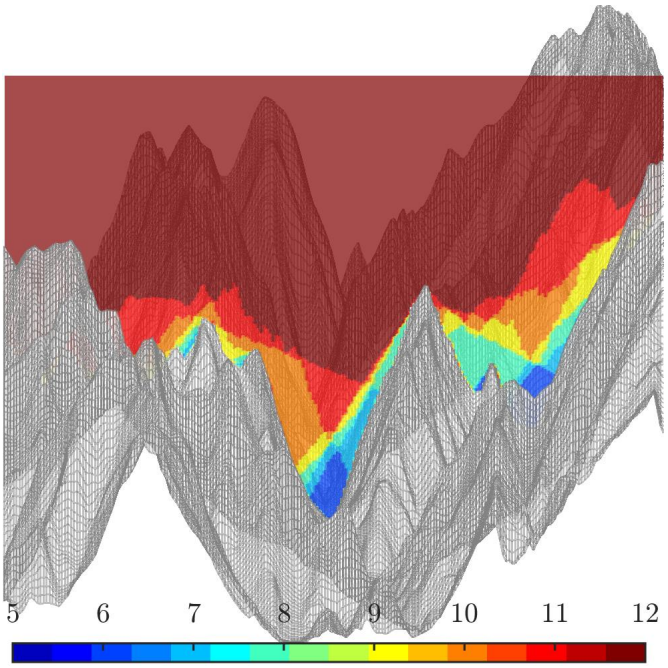


Fig. 6. 2D section of the control volume showing the number of visible satellites

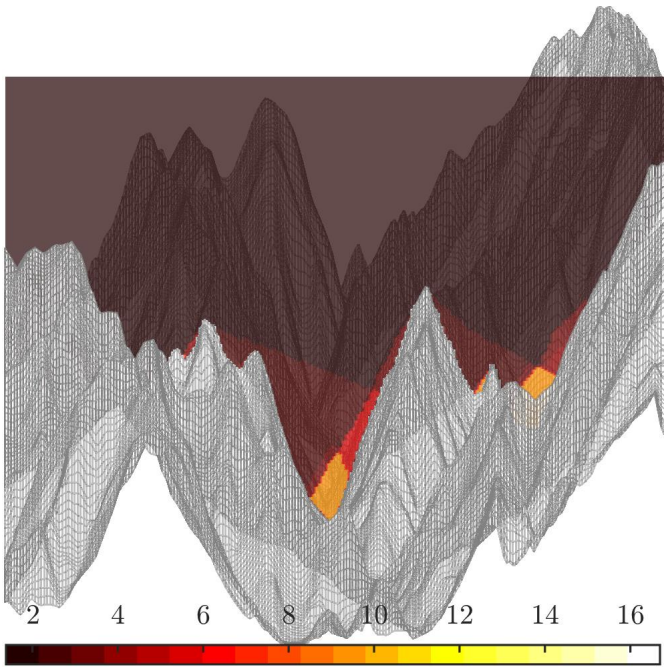


Fig. 7. 2D section of the control volume showing the value of the DOP

The value of θ_{lim} that defines the visibility cone aperture is $\theta_{lim} = 85\,\mathrm{deg}$. The results proposed in this paper have been obtained using the software MATLAB on an Intel Core i7-4710HQ 64 bit, 2.50 GHz with 16 GB RAM.

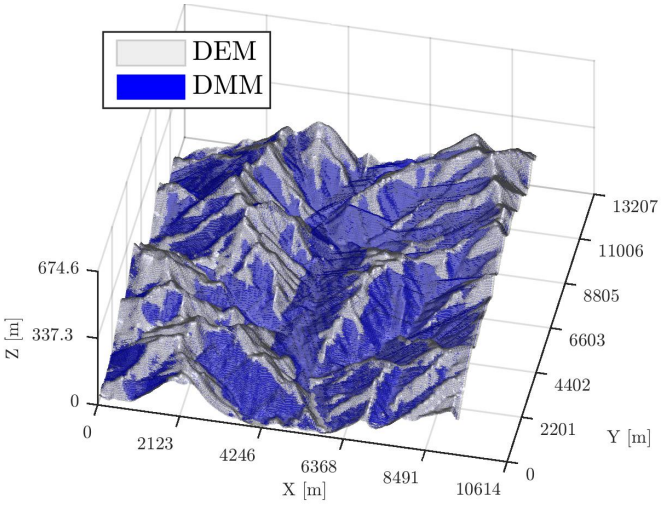


Fig. 8. DMM over the entire DEM relative to $DOP_{lim} = 2.5$ for the situation of Fig. 6 and Fig. 7

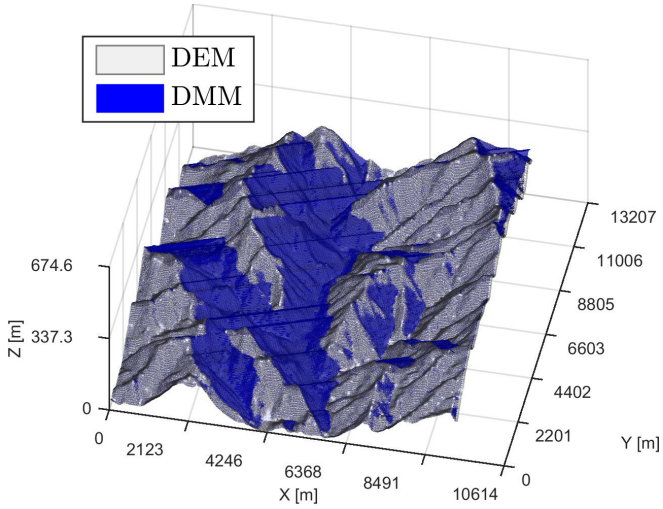


Fig. 9. DMM over the entire DEM relative to $DOP_{lim} = 2.5$ with a worse satellite geometry

A. DMM results

In Fig. 6 the DEM with a 2D section of the control volume is shown. The simulated Global Positioning System (GPS) almanac data refer to the 02:46 pm (CET) of February the 22^{th} 2017. The theoretical results presented graphically in Fig. 3 are confirmed by the simulation. The colormap indicates the number of visible satellites in any point of the control volume. As expected, the minimum number of visible satellites is registered in the valley with only 6 visible satellites over a maximum of 12 in the upper part of the control volume. In Fig. 7 the control volume colormap indicates the values of the DOP relative to the visible satellites. As expected, the highest value of the DOP corresponds to the region where the number of visible satellites has a minimum (blue region in Fig. 6). Considering that the number of satellites involved in

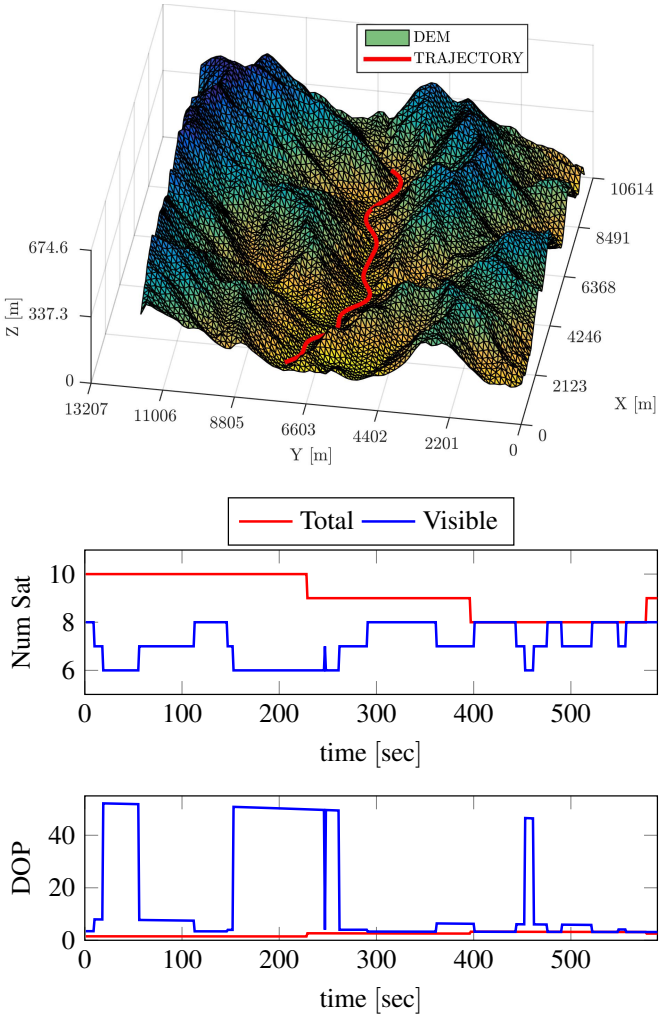


Fig. 10. Digital Elevation Model with 3D UAV trajectory profile in red. The second figure shows the total number of satellites (red), and number the of visible satellites (blue) along the trajectory. The third plot shows the DOP along the trajectory for total (red) and visible (blue) satellites

the calculation of the DOP is less influential than the satellite geometry, the DOP map in the control volume has a different geometry compared to that of Fig. 7. This means that some of the invisible satellites have a very low contribution in the DOP value and the colormap resolution is not sufficient to capture this variation.

In Fig. 8 the DEM with a superposition of the DMM is shown. The DMM has been calculated using a value of $DOP_{lim} = 2.5$. From the definition of $\mathbf{P}_{NFZ}$, all the points below the DMM have a DOP value higher than the selected threshold. This means that for path planning purposes this virtual DEM should be used instead of the real one. A more critical situation is shown in Fig. 9, where, because of the bad geometry of the visible satellites, the value of the DOP in the valley increases everywhere above the threshold, eliminating the entire region from the flight zone. The satellite geometry relative to this situation comes from a different simulation time with respect to the results of Fig. 6, 7 and 8.

B. SBOA results

The algorithm is applied to the simulated trajectory shown in Fig. 10. The trajectory refers to the 15^{th} of November 2016 at 07:30 pm (CET). Fig. 10 also reports the total number of satellites in the visibility cone and the effective number of visible satellites in each point of the trajectory. Due to the occlusion of some satellites, the value of the DOP is subject to abrupt variations. This figure validates also the statement of Section II-C about the constancy of the DOP for relatively small spatial variations. The simulation time required to process the entire trajectory is 1.16 sec. Using the machine specified in Section IV the system runs at about 500 Hz. The dynamical model assumed in the Kalman filter for the UAV motion is a constant velocity system. The discrete time state space equation is described by the following linear relation

$$\mathbf{x}_n = \mathbf{F} \cdot \mathbf{x}_{n-1} \tag{28}$$

where the state space vector is

$$\mathbf{x} = \begin{bmatrix} x & \dot{x} & y & \dot{y} & z & \dot{z} & C_b & C_d \end{bmatrix}^T \tag{29}$$

where x, y and z the NED coordinates and C_b and C_d are the clock error and the clock error drift. The state transition matrix $\mathbf{F}$ is defined as follows

$$\mathbf{F} = \begin{bmatrix} \overbrace{\begin{bmatrix} 1 & \Delta t \\ 0 & 1 \end{bmatrix}}^{\mathbf{F_1}} & \mathbf{0}_{2\times2} & \mathbf{0}_{2\times2} & \mathbf{0}_{2\times2} \\ \mathbf{0}_{2\times2} & \mathbf{F_1} & \mathbf{0}_{2\times2} & \mathbf{0}_{2\times2} \\ \mathbf{0}_{2\times2} & \mathbf{0}_{2\times2} & \mathbf{F_1} & \mathbf{0}_{2\times2} \\ \mathbf{0}_{2\times2} & \mathbf{0}_{2\times2} & \mathbf{0}_{2\times2} & \mathbf{F_1} \end{bmatrix}, \tag{30}$$

where Δt is the integration time step. The system noise covariance matrix $\mathbf{Q}$ is

$$\mathbf{Q} = \begin{bmatrix} q_x \overbrace{\begin{bmatrix} \frac{\Delta t^3}{3} & \frac{\Delta t^2}{2} \\ \frac{\Delta t^2}{2} & \Delta t \end{bmatrix}}^{\mathbf{Q_1}} & \mathbf{0}_{2\times2} & \mathbf{0}_{2\times2} & \mathbf{0}_{2\times2} \\ \mathbf{0}_{2\times2} & q_y\mathbf{Q_1} & \mathbf{0}_{2\times2} & \mathbf{0}_{2\times2} \\ \mathbf{0}_{2\times2} & \mathbf{0}_{2\times2} & q_z\mathbf{Q_1} & \mathbf{0}_{2\times2} \\ \mathbf{0}_{2\times2} & \mathbf{0}_{2\times2} & \mathbf{0}_{2\times2} & q_c\mathbf{Q_1} \end{bmatrix} \tag{31}$$

where q_x, q_y, q_z and q_c are the noise Power Spectral Densities (PSD) along each Cartesian direction and for the clock error. The non-linear measurement model is made by the pseudoranges between the visible satellites and the on-board receiver

$$\hat{\mathbf{z}} = \mathbf{h}(\mathbf{x}_n) = \begin{bmatrix} \hat{\rho}_1 & \hat{\rho}_2 & \dots & \hat{\rho}_{N_s} \end{bmatrix}^T. \tag{32}$$

The pseudorange $\hat{\rho}_i$ is defined as follows

$$\hat{\rho}_{i=1\dots N_s} = \rho_{i=1\dots N_s} + C_b \tag{33}$$

where N_s is the number of satellites and ρ_i is defined in Eq. 18. The measurement matrix $\mathbf{H}$ is the Jacobian of $\mathbf{h}(\mathbf{x}_n)$ and is defined as follows

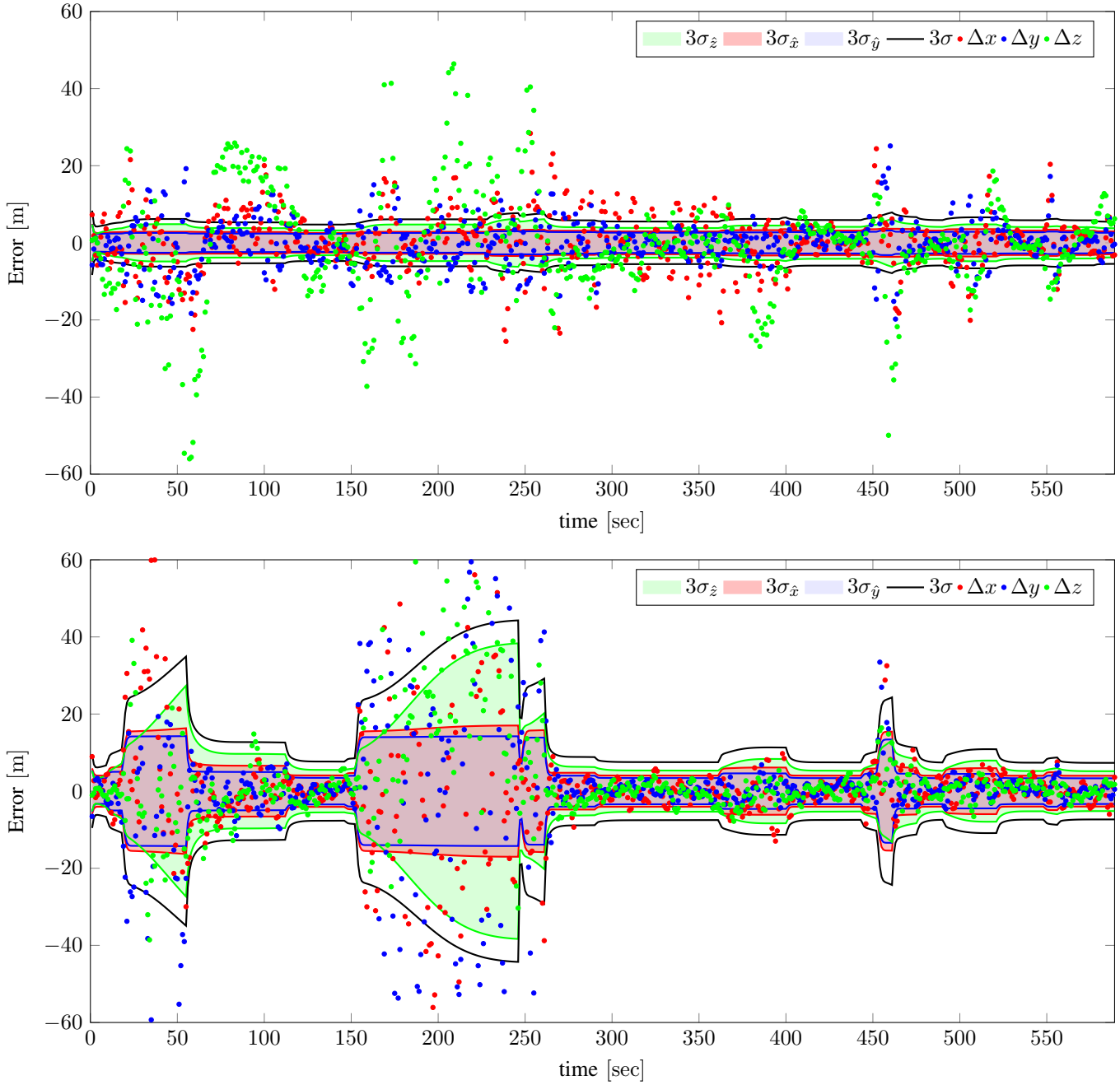


Fig. 11. Error of the estimated position with respect to the real one, for each Cartesian coordinate (Δx, Δy and Δz). The coloured area is the 3σ confidence interval for each coordinate. The black solid line is the confidence interval relative to the norm of the three components $\sigma = \sqrt{\sigma_{\hat{x}}^2 + \sigma_{\hat{y}}^2 + \sigma_{\hat{z}}^2}$. The latter is used to define a spherical Safety Bubble. The upper figure presents the results relative to the weak solution. The lower figure shows the results for the proposed approach

$$\mathbf{H} = \begin{bmatrix} \frac{\partial \hat{\rho}_1}{\partial x} & \frac{\partial \hat{\rho}_1}{\partial x}\Delta t & \cdots & \frac{\partial \hat{\rho}_1}{\partial z} & \frac{\partial \hat{\rho}_1}{\partial z}\Delta t & 1 & 0 \\ \frac{\partial \hat{\rho}_2}{\partial x} & \frac{\partial \hat{\rho}_2}{\partial x}\Delta t & \cdots & \frac{\partial \hat{\rho}_2}{\partial z} & \frac{\partial \hat{\rho}_2}{\partial z}\Delta t & 1 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots \\ \frac{\partial \hat{\rho}_{N_s}}{\partial x} & \frac{\partial \hat{\rho}_{N_s}}{\partial x}\Delta t & \cdots & \frac{\partial \hat{\rho}_{N_s}}{\partial z} & \frac{\partial \hat{\rho}_{N_s}}{\partial z}\Delta t & 1 & 0 \end{bmatrix} \quad (34)$$

where the derivatives are calculated as

$$\frac{\partial \hat{\rho}_i}{\partial \xi} = -\frac{\xi_{S_i} - \xi}{\rho_i} \quad (35)$$

with $\xi = x, y, z$. The signal accuracy can be expressed as a function of different error sources

$$\sigma_\rho = \sigma_\rho \begin{pmatrix} \delta_{\rho,clock} & \delta_{\rho,atm} & \delta_{\rho,rec} \end{pmatrix} \quad (36)$$

where $\delta_{\rho,clock}$, $\delta_{\rho,atm}$ and $\delta_{\rho,rec}$ are respectively the clock, tropospheric and receiver errors. Clock and receiver errors

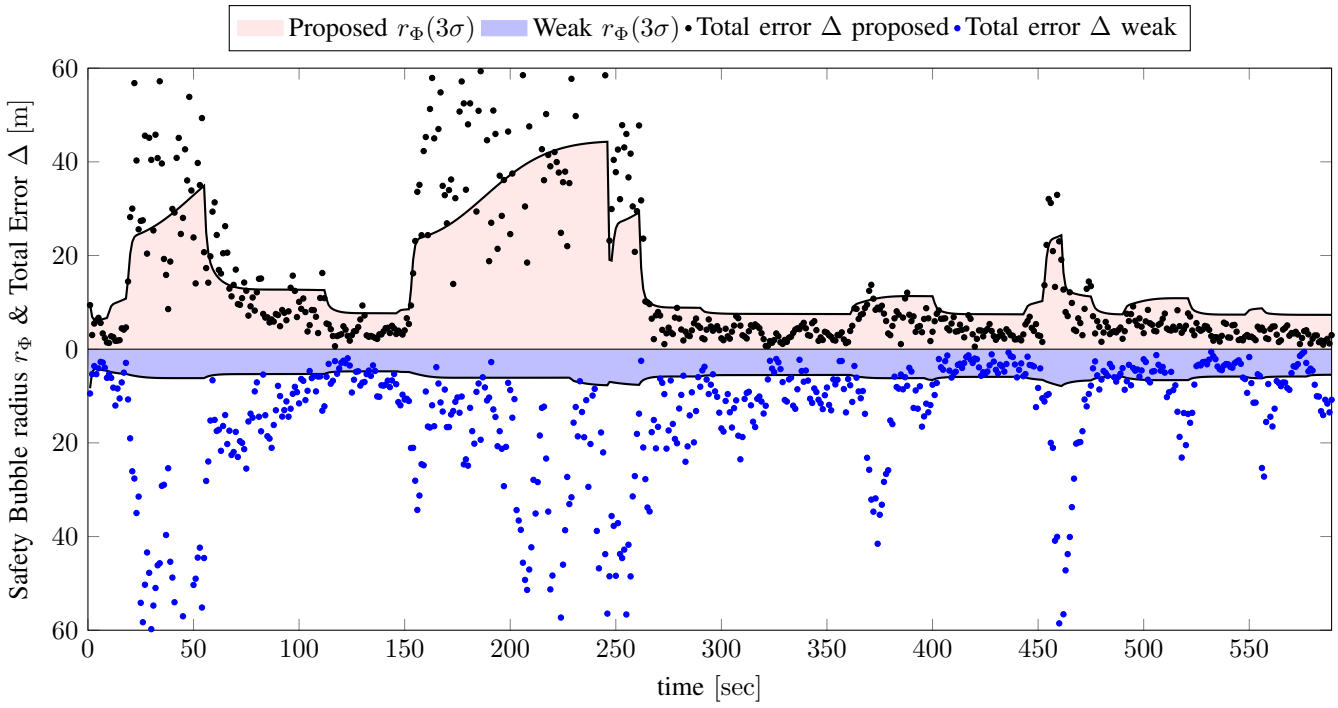


Fig. 12. Safety Bubble radius $r_\Phi(3\sigma)$ relative to the proposed approach and safety bubble radius for the weak solution. The dotted solution is the total error $\Delta = \sqrt{\Delta x^2 + \Delta y^2 + \Delta z^2}$ of the estimated position for the proposed and weak solution

are modelled as a zero mean white noise, whereas $\delta_{\rho,atm}$ is obtained by means of Niell Mapping Function

$$\delta_{\rho,atm} = T_{dry}(z_{NED})M_{dry}(E) + T_{wet}M_{wet}(E). \quad (37)$$

E is the satellite elevation, whereas T_{dry} and T_{wet} represent respectively dry and wet delay at altitude z_{NED}. For a complete formulation of T_{dry}, T_{wet}, M_{dry} and M_{wet} refer to [42], [43]. The multiple reflected signal has been simulated by using the approach proposed in [44]. In this case, by knowing which are the non visible satellite, their position, the receiver position and the DEM, with the hypothesis of single reflection, the signal delay has been calculated and used to simulate the noisy measurements. Considering that the simulation is carried out in a mountain canyon, a signal reflected by the side of a mountain has to travel at least half width of the canyon before reaching the receiver antenna. This justifies the high error in Fig. 11 and 12 in the presence of non-visible satellites. The measurement covariance matrix $\mathbf{R}$ is defined in Eq. 26 and is dynamically updated, based on the local value of DOP.

To assess the method performance, in this section the proposed method is compared with a *weak solution* technique, which is obtained with the hypothesis that all the received measurements are taken into account. In this situation, the measurements relative to the satellites that are in NLOS are affected by multiple reflections before reaching the receiving antenna. The proposed method uses the value of the DOP, obtained only by taking into account the visible satellites, to scale the value of the components of the matrix $\mathbf{R}$. The *weak solution*, on the contrary, uses the value of the DOP associated to all the satellites in the visibility cone and scales the components of the matrix $\mathbf{R}$ by choosing a higher value of the multiplicative coefficient μ_i for the satellites that are not in line of sight. In Fig. 11 the outcome provided by the weak solution (upper) and the proposed approach (lower) is presented. The scatter solution is the error of the estimated position coordinates with respect to the real platform position. The coloured regions are the 3σ confidence intervals for each Cartesian component. The black solid line indicates the 3σ confidence intervals for the norm of the three single components $\sigma = \sqrt{\sigma_{\hat{x}}^2 + \sigma_{\hat{y}}^2 + \sigma_{\hat{z}}^2}$. This value is used to calculate the Safety Bubble $\mathbf{\Phi}$. From the results of the weak solution, it is clear that the system underestimates the uncertainty associated to the estimated position. The high number of outliers, in fact, confirms the weakness of this approach that uses, for the position estimation, measures affected by multiple reflections. The safety bubble associated to this approach is called $\mathbf{\Phi_w}$. The method proposed in this paper generates a more reactive confidence interval capable of accommodating abrupt variations in the measurement accuracy. In Fig. 12 the Safety Bubble radius r_ϕ along the trajectory and the total error $\Delta = \sqrt{\Delta x^2 + \Delta y^2 + \Delta z^2}$ for the weak and proposed solution are presented. The value of Δ is the distance between the true and the estimated solution. The upper part of this figure is relative to the proposed solution, whereas the lower part is the safety bubble radius associated to the weak solution. The weak solution provides an estimation of the platform position that is inside the Safety Bubble with a probability of $34\,\%$, whereas the Safety Bubble generated by the proposed method contains the position estimation with a probability of $76.4\,\%$. The exclusion of NLOS satellites and the use of a dynamic covariance matrix $\mathbf{R}$ can guarantee a more

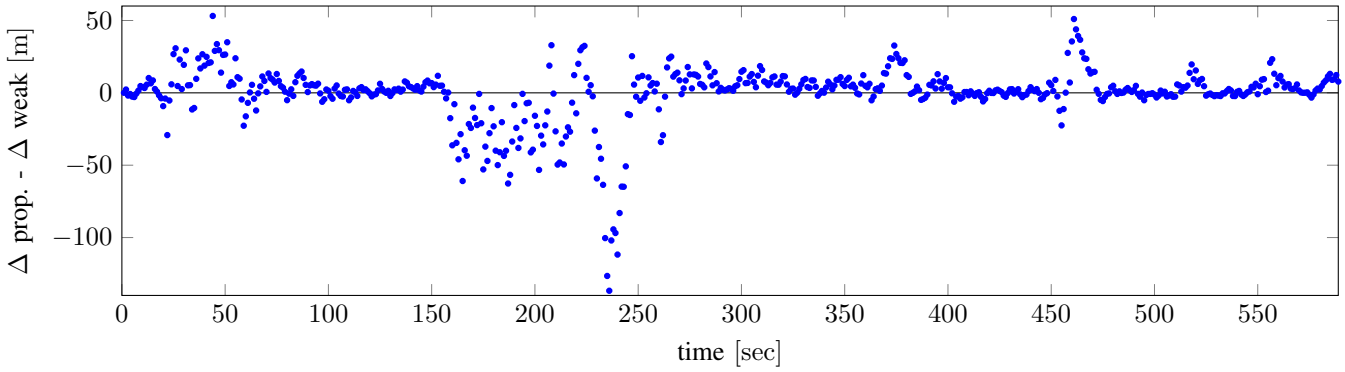


Fig. 13. Difference between the total error Δ obtained with the weak solution and the total error provided by the proposed method. For the 63 % of time this difference is positive

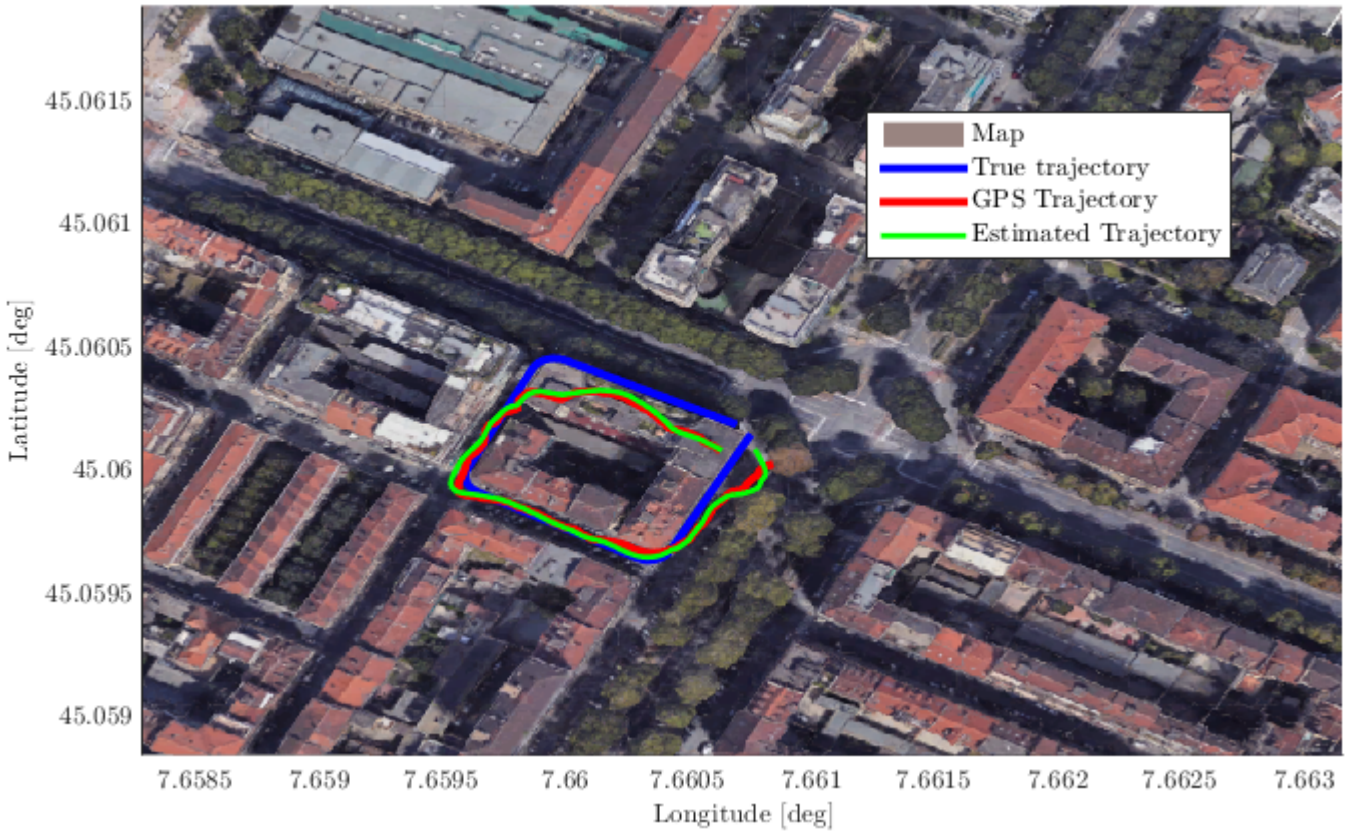


Fig. 14. Comparison of the true, GPS and estimated trajectory. Map image: Google Earth Pro.

accurate solution and provide, as mathematically demonstrated in Section III, a reactive Safety Bubble capable of indicating which is the minimum safety distance to maintain from the DEM. Intuitively, the higher accuracy of the solution can be seen as a consequence of the exclusion of corrupted measurements from the filtering process. The inclusion of these measurements, as in the case of the weak solution, degrades the position estimation accuracy even if the gain of the Kalman filter associated to the corrupted measurements is very low (refer to TABLE III). This concept is confirmed by the numerical results, where for the 63 % of time the total error of the proposed method is lower than the one provided by the weak solution. The difference between the total error of the weak solution and the one of the proposed method is shown in Fig. 13.

C. Experimental results

The algorithm has been validated by performing a field experiment. The experiment was performed during daytime in the city centre of Turin, Italy. The test consisted in taking GPS measurements using a GPS receiver in an urban canyon. During the experiment a small unmanned vehicle was operated around a building, surrounded, on two sides, by buildings with an height of 30 m. Fig. 14 shows the real trajectory (blue), the measured trajectory (red) and the estimated one (green). The

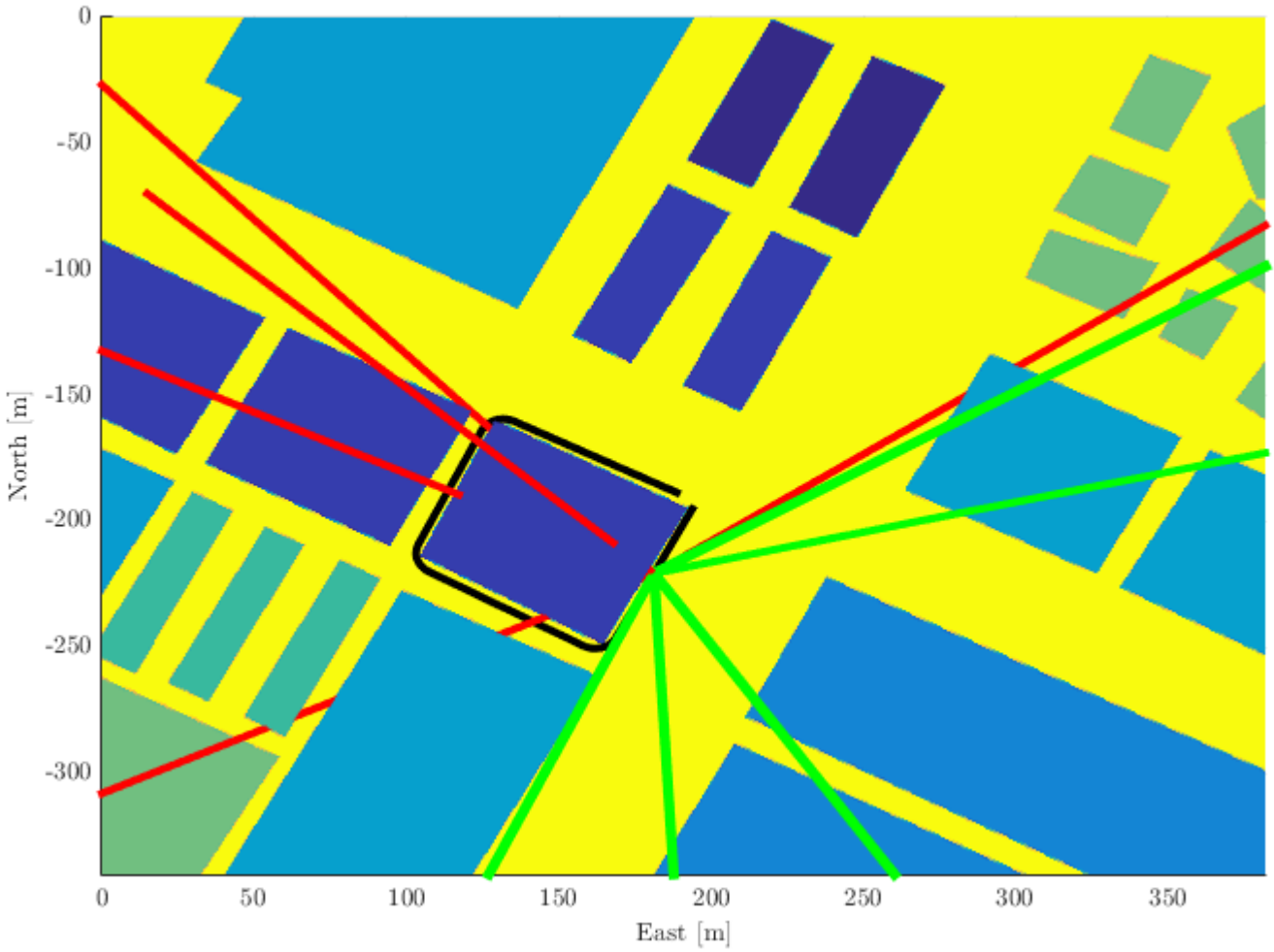


Fig. 15. DEM with visible (green) and non-visible (red) satellite-platform directions at 20 sec from the beginning of the experiment. The real trajectory is black

linear Kalman filter is used to estimate the platform position regardless of the EKF used for the simulation in the previous section. This condition is driven by the availability of the GPS positions instead of the pseudoranges. Fig. 15 shows the DEM with the line connecting the satellites and the platform position at 20 sec from the beginning of the experiment. The lines associated to the visible satellites are green, whereas those associated to the occluded ones are red. Fig. 16 shows the number of visible satellites, the DOP along the trajectory and the position errors. The 1σ safety bubble is shown in the third plot of Fig. 16. The total position error is contained in the safety bubble for the $63.7\,\%$ of the experiment for the 1σ condition. This percentage rises to the $94.6\,\%$ if the 2σ safety bubble is used.

V. Conclusions

This paper presents a very simple and efficient algorithm ($\mathcal{O}(n)$) to determine the invisible regions of a generic 3D surface, when observed from a given point of view. In this paper the algorithm is used to calculate the number of visible satellites in a control volume and the consequent DOP distribution. This information is used to calculate a dynamic DEM for path planning purposes and the Safety Bubble that indicates which is the minimum distance that should be maintained from the DEM. To assess its performance, the proposed method has been compared with a weak solution technique. The experimental results, obtained for a challenging mountainous scenario, show that the safety bubble dimension ranges from $10\,\mathrm{m}$ to $50\,\mathrm{m}$. In this condition the probability that the safety bubble contains the platform is above $70\,\%$, which is a clear improvement with respect to the weak solution. The algorithm is conceived to be easily integrated with already certified and operating navigation systems, as the terrain awareness and warning system (TAWS). In addition, any technique developed to enhance the navigation in GNSS-denied spaces can be easily integrated with this system. Potential algorithm implications, like the SBOA system have been tested so as to highlight the generality of the approach and its suitability for applications involving the autonomous navigation.

Funding

This research did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors.

Declaration of conflict of interest

The authors declare that there are no conflicts of interest.

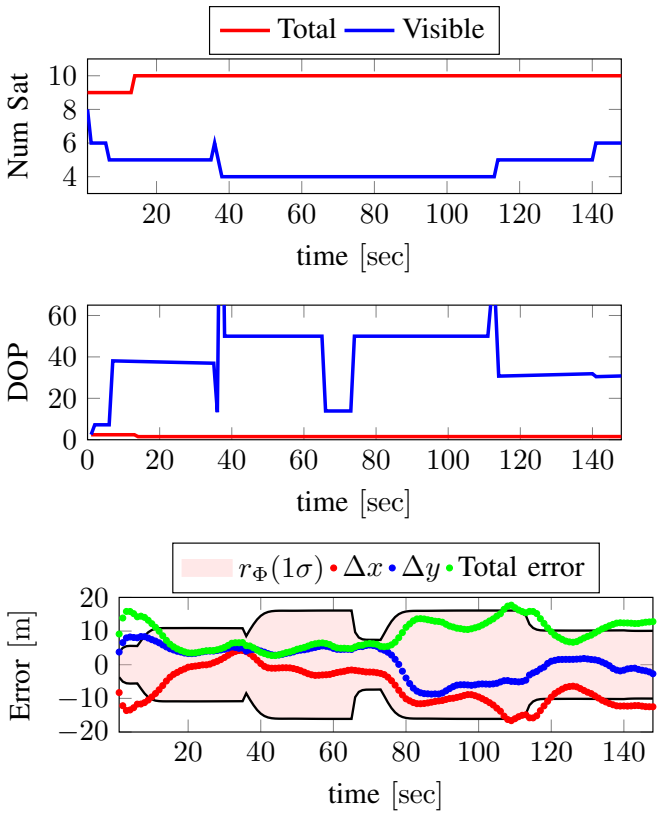


Fig. 16. The first figure shows the total number of satellites (red), and number the of visible satellites (blue) along the trajectory. The second plot shows the DOP along the trajectory for total (red) and visible (blue) satellites. The third figure shows the safety bubble radius and the error of the Cartesian coordinates with respect to the true trajectory

REFERENCES

[1] P. D. Groves, *Principles of GNSS, inertial, and multisensor integrated navigation systems*. Artech house, 2013.

[2] J. Moreau, S. Ambellouis, and Y. Ruichek, "Fisheye-based method for gps localization improvement in unknown semi-obstructed areas," *Sensors*, vol. 17, no. 1, p. 119, 2017. [Online]. Available: http://dx.doi.org/10.3390/s17010119

[3] F. Peyret, D. Bétaille, P. Carolina, R. Toledo-Moreo, A. F. Gómez-Skarmeta, and M. Ortiz, "Gnss autonomous localization: Nlos satellite detection based on 3-d maps," *IEEE Robotics & Automation Magazine*, vol. 21, no. 1, pp. 57–63, 2014. [Online]. Available: http://dx.doi.org/10.1109/MRA.2013.2295944

[4] M. Obst, S. Bauer, and G. Wanielik, "Urban multipath detection and mitigation with dynamic 3d maps for reliable land vehicle localization," in *Position Location and Navigation Symposium (PLANS), 2012 IEEE/ION*. IEEE, 2012, pp. 685–691. [Online]. Available: http://dx.doi.org/10.1109/PLANS.2012.6236944

[5] A. S. Glassner, *An introduction to ray tracing*. Elsevier, 1989.

[6] A. Kumar, Y. Sato, T. Oishi, and K. Ikeuchi, "Identifying reflected gps signals and improving position estimation using 3d map simultaneously built with laser range scanner," *Rapport technique, Computer Vision Laboratory, Institute of Industrial Science, The University of Tokyo*, 2014.

[7] J. Marais, C. Meurie, D. Attia, Y. Ruichek, and A. Flancquart, "Toward accurate localization in guided transport: combining gnss data and imaging information," *Transportation Research Part C: Emerging Technologies*, vol. 43, pp. 188–197, 2014. [Online]. Available: http://dx.doi.org/10.1016/j.trc.2013.11.008

[8] D. Attia, "Segmentation d'images par combinaison adaptative couleur-texture et classification de pixels.: Applications à la caractérisation de l'environnement de réception de signaux gnss," Ph.D. dissertation, Université de Technologie de Belfort-Montbeliard, 2013.

[9] J. Levinson, M. Montemerlo, and S. Thrun, "Map-based precision vehicle localization in urban environments." in *Robotics: Science and Systems*, vol. 4. Citeseer, 2007, p. 1.

[10] S. Weiss, D. Scaramuzza, and R. Siegwart, "Monocular-slam–based navigation for autonomous micro helicopters in gps-denied environments," *Journal of Field Robotics*, vol. 28, no. 6, pp. 854–874, 2011. [Online]. Available: http://dx.doi.org/10.1002/rob.20412

[11] A. S. Huang, A. Bachrach, P. Henry, M. Krainin, D. Maturana, D. Fox, and N. Roy, "Visual odometry and mapping for autonomous flight using an rgb-d camera," in *Robotics Research*. Springer, 2017, pp. 235–252. [Online]. Available: http://dx.doi.org/10.1007/978-3-319-29363-9_14

[12] A. Bachrach, S. Prentice, R. He, and N. Roy, "Range–robust autonomous navigation in gps-denied environments," *Journal of Field Robotics*, vol. 28, no. 5, pp. 644–666, 2011. [Online]. Available: http://dx.doi.org/10.1002/rob.20400

[13] G. Chowdhary, E. N. Johnson, D. Magree, A. Wu, and A. Shein, "Gps-denied indoor and outdoor monocular vision aided navigation and control of unmanned aircraft," *Journal of Field Robotics*, vol. 30, no. 3, pp. 415–438, 2013. [Online]. Available: http://dx.doi.org/10.1002/rob.21454

[14] D. Stepanov and I. Tishchenko, "Methods of visual navigation of the uav flying over the nonplanar district," in *Robot Intelligence Technology and Applications 4*. Springer, 2017, pp. 551–561. [Online]. Available: http://dx.doi.org/10.1007/978-3-319-31293-4_45

[15] M. Achtelik, A. Bachrach, R. He, S. Prentice, and N. Roy, "Stereo vision and laser odometry for autonomous helicopters in gps-denied indoor environments," in *SPIE Defense, security, and sensing*. International Society for Optics and Photonics, 2009, pp. 733 219–733 219. [Online]. Available: http://dx.doi.org/10.1117/12.819082

[16] R. Kapoor, S. Ramasamy, A. Gardi, C. Bieber, L. Silverberg, and R. Sabatini, "A novel 3d multilateration sensor using distributed ultrasonic beacons for indoor navigation," *Sensors*, vol. 16, no. 10, p. 1637, 2016. [Online]. Available: https://doi.org/10.3390/s16101637

[17] X. Zhong, R. Rabiee, Y. Yan, and W. P. Tay, "A particle filter for vehicle tracking with lane level accuracy under gnss-denied environments," in *2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSC)*, 10 2017, pp. 1–6. [Online]. Available: https://doi.org/10.1109/ITSC.2017.8317659

[18] S. Kanaki, M. Hashimoto, Y. Yoden, and K. Takahashi, "Laser-based cooperative tracking of vehicles and people by multiple mobile robots in gnss-denied environments," in *2017 IEEE International Conference on Advanced Intelligent Mechatronics (AIM)*, 7 2017, pp. 1228–1233. [Online]. Available: https://doi.org/10.1109/AIM.2017.8014186

[19] V. Indelman, "Cooperative multi-robot belief space planning for autonomous navigation in unknown environments," *Autonomous Robots*, pp. 1–21, 2017. [Online]. Available: http://dx.doi.org/10.1007/s10514-017-9620-6

[20] S. Hening, C. A. Ippolito, K. S. Krishnakumar, V. Stepanyan, and M. Teodorescu, "3d lidar slam integration with gps/ins for uavs in urban gps-degraded environments," in *AIAA Information Systems-AIAA Infotech@ Aerospace*. AIAA, 2017, p. 0448.

[21] K. Gryte, J. M. Hansen, T. Johansen, and T. I. Fossen, "Robust navigation of uav using inertial sensors aided by uwb and rtk gps," in *AIAA Guidance, Navigation, and Control Conference*. AIAA, 2017, p. 1035.

[22] F. Gandor, M. Rehak, and J. Skaloud, "Photogrammetric mission planner for rpas," *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. XL-1/W4, pp. 61–65, 2015.

[23] F. Zimmermann, C. Eling, L. Klingbeil, and H. Kuhlmann, "Precise positioning of uavs–dealing with challenging rtk-gps measurement conditions during automated uav flights," *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. 4, p. 95, 2017.

[24] Zhang, Guohao, Hsu, and Li-Ta, "A new path planning algorithm based on gnss localization error map," in *Proceedings of the 30th International Technical Meeting of The Satellite Division of the Institute of Navigation*, 9 2017, pp. 637–654.

[25] S. Peyraud, D. Btaille, S. Renault, M. Ortiz, F. Mougel, D. Meizel, and F. Peyret, "About non-line-of-sight satellite detection and exclusion in a 3d map-aided localization algorithm," *Sensors*, vol. 13, no. 1, pp. 829–847, 2013. [Online]. Available: http://www.mdpi.com/1424-8220/13/1/829

[26] P. D. Groves, Z. Jiang, L. Wang, and M. K. Ziebart, "Intelligent urban positioning using multi-constellation gnss with 3d mapping and nlos signal detection," in *Proceedings of ION GNSS 2012*, 2012.

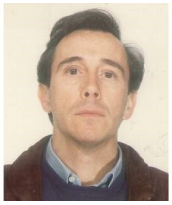
[27] L. Wang, P. D. Groves, and M. K. Ziebart, "Gnss shadow matching: Improving urban positioning accuracy using a 3d city model with optimized visibility scoring scheme," *Navigation*, vol. 60, no. 3, pp. 195–207, 2013. [Online]. Available: http://dx.doi.org/10.1002/navi.38
[28] M. Adjrad and P. D. Groves, "Enhancing conventional gnss positioning with 3d mapping without accurate prior knowledge," in *Proceedings of the 28th International Technical Meeting of The Satellite Division of the Institute of Navigation*. The Institute of Navigation, 2015.
[29] L. Wang, P. D. Groves, and M. K. Ziebart, "Urban positioning on a smartphone: Real-time shadow matching using gnss and 3d city models," in *Proceedings of the 26th International Technical Meeting of The Satellite Division of the Institute of Navigation*. The Institute of Navigation, 2013.
[30] ——, "Smartphone shadow matching for better cross-street gnss positioning in urban environments," *The Journal of Navigation*, vol. 68, no. 3, pp. 411–433, 2015. [Online]. Available: http://dx.doi.org/10.1017/S0373463314000836
[31] F. De Vivo, M. Battipede, and P. Gili, "Occlusion points identification algorithm," *Computer-Aided Design*, vol. 91, no. Supplement C, pp. 75 – 83, 2017. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0010448517301185
[32] H. Oliveira, A. Habib, A. Dal Poz, and M. Galo, "Height gradient approach for occlusion detection in uav imagery," *The International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. 40, no. 1, p. 263, 2015. [Online]. Available: http://dx.doi.org/10.5194/isprsarchives-XL-1-W4-263-2015
[33] F. De Vivo, M. Battipede, P. Gili, A. J. Yezzi, E. Feron, and E. Johnson, "Real-time fire segmentation via active contours for uav integrated wildfire propagation prediction," in *2018 AIAA Information Systems-AIAA Infotech@ Aerospace*. AIAA, 2018, p. 1488. [Online]. Available: https://doi.org/10.2514/6.2018-1488
[34] F. De Vivo, "Uav application in wildfires," YouTube, 2 2018. [Online]. Available: https://www.youtube.com/watch?v=5Jwd_90JpbA&list=PLBVpcsfm--mZ8sIiqjTdJxXG-GEb0Wdza
[35] T. Kelso. CelesTrak. [Online]. Available: https://celestrak.com/
[36] M. Dunn, *Interface specification IS-GPS-200*, Global Positioning Systems Directorate, 2013. [Online]. Available: http://www.gps.gov/technical/icwg/IS-GPS-200H.pdf
[37] S. Cakaj, B. Kamo, A. Lala, and A. Rakipi, "The coverage analysis for low earth orbiting satellites at low elevation," *Int. J. Adv. Comput. Sci. Appl.(ijacsa)*, vol. 5, no. 6, 2014.
[38] T. Möller and B. Trumbore, "Fast, minimum storage ray/triangle intersection," in *ACM SIGGRAPH 2005 Courses*. ACM, 2005, p. 7.
[39] R. B. Langley *et al.*, "Dilution of precision," *GPS world*, vol. 10, no. 5, pp. 52–59, 1999.
[40] F. De Vivo, A. Brandl, M. Battipede, and P. Gili, "Joseph covariance formula adaptation to square-root sigma-point kalman filters," *Nonlinear Dynamics*, pp. 1–18, 2017. [Online]. Available: http://dx.doi.org/10.1007/s11071-017-3356-x
[41] F. De Vivo, M. Battipede, P. Gili, and A. Brandl, "Ill-conditioned problems improvement adapting joseph covariance formula to non-linear bayesian filters," *WSEAS Transaction on Electronics*, vol. 7, pp. 18–25, 2016. [Online]. Available: http://10.13140/RG.2.1.3027.0960
[42] A. Niell, "Global mapping functions for the atmosphere delay at radio wavelengths," *Journal of Geophysical Research: Solid Earth*, vol. 101, no. B2, pp. 3227–3246, 1996.
[43] A. Brandl, M. Battipede, P. Gili, and A. Lerro, "Sensitivity analysis of a neural network based avionic system by simulated fault and noise injection," in *2018 AIAA Modeling and Simulation Technologies Conference*, ser. AIAA SciTech Forum. American Institute of Aeronautics and Astronautics, Jan 2018, 0. [Online]. Available: https://doi.org/10.2514/6.2018-0122
[44] P. Pisova and J. Chod, "Detection of gnss signals propagation in urban canyos using 3d city models," *Advances in Electrical and Electronic Engineering*, vol. 13, no. 1, 2015. [Online]. Available: http://advances.utc.sk/index.php/AEEE/article/view/1291



Francesco De Vivo was born in Salerno, Italy in 1990. He received the B.S. and M.S. in aerospace engineering from the University of Naples Federico II, Italy, respectively in 2012 and 2015. He is actually a PhD student at Polytechnic of Turin, Italy, and visiting scholar at Georgia Institute of Technology, Atlanta. He is main author of several published works on advanced Bayesian filtering techniques for filter stability improvement and computer vision applications. His current research activity is focused on the development of data-driven wildfire simulators using Level Set techniques and advanced image segmentation approaches. His research interests include also multi sensor data fusion techniques, UAV navigation in GNSS-denied environment, aircraft sense and avoid and flight simulators development.



Manuela Battipede got her PhD in aerospace engineering in 2000 at the Polytechnic of Turin, where she is currently employed as Associate Professor in the flight mechanic field, teaching M.S. courses "aerospace flight mechanics", "aircraft guidance and control" and co-teaching the B.S. courses "fundamental in flight mechanics". After a period abroad at the West Virginia University, where she has worked as a Visiting Assistance Professor in the Mechanical and Space Department, she has collaborated with the major aeronautical Italian companies. She is co-author of more than 60 international journal and conference papers in the areas of atmospheric flight mechanics, modelling & simulation as well as guidance navigation & control aircraft for fixed and rotary-wing aircraft. She is a reviewer for numerous international journals and conferences.



Piero Gili has been doing research work at the Department of Aeronautical and Space Engineering in the flight mechanics sector since 1983, when he was appointed Researcher. In 1992 he held an appointment in flight test techniques, and one in flight dynamics in 1997. His research activity mostly focuses on fixed and rotating wing aircraft dynamics and control, satellite and lighter than-air-aircraft dynamics and control. His activity is supported by over 100 publications in international congresses. The development of innovative control systems is the main subject of his research activity. He was Visiting Research Associate Professor at MAE (Department of Mechanical and Aerospace Engineering), West Virginia University, working in the fault tolerant control systems area. In particular, the invitation was aimed at developing a partnership on the NASA F-15 Intelligent Flight Control System (IFCS) project. He also holds a patent for the project "High-manoeuvrability aerostatic-type airship", Elica o rotore a passo variabile con svergolamento variabile di rendimento ottimo and "System and process for measuring and evaluating air and inertial data".