Terrain-Aided SLAM with Limited-Size Reference Maps Using Gaussian Processes

PALMIER Camille
CESTA, DAM, CEA
33 114 Le Barp, FRANCE
camille.palmier@cea.fr

GIREMUS Audrey
IMS (Université de Bordeaux, CNRS, Bordeaux INP)
33 400 Talence, France
audrey.giremus@u-bordeaux.fr

MINVIELLE Pierre
CESTA, DAM, CEA
33 114 Le Barp, FRANCE
pierre.minvielle@cea.fr

VACAR Cornelia
CESTA, DAM, CEA
33 114 Le Barp, FRANCE
cornelia.vacar@cea.fr

BOURMAUD Guillaume
IMS (Université de Bordeaux, CNRS, Bordeaux INP)
33 400 Talence, France
guillaume.bourmaud@u-bordeaux.fr

***Abstract*—Terrain-aided navigation is often used for unmanned aerial vehicles. This method consists in estimating the current dynamics and position of a vehicle by matching terrain profiles obtained by exteroceptive sensors with onboard maps. Whereas the resolution of the map plays a crucial role in minimising positioning errors, accurate reference maps are difficult to upload when transmission restrictions are experienced. In this context of sparse communications, we propose to model the maps by Gaussian Processes completely characterised by a mean function that can be interpreted as a low resolution approximation of the map and a covariance kernel to account for the spatial correlations. The objective of this paper is then to perform simultaneous localisation and mapping by leveraging radio-altimetric data. The inference is carried out by a non-linear Bayesian filter that takes advantage of the conditional linear Gaussianity of the state space model: the Rao-Blackwellised particle filter.**

***Index Terms*—Terrain-Aided SLAM, Gaussian Processes, Particle Filtering**

I. Introduction

In GPS-denied navigation [1], unmanned aerial vehicles (UAV) only take advantage of onboard sensors. Terrain-Aided Navigation (TAN) consists in estimating the current dynamics and position of a vehicle by matching terrain profiles obtained by exteroceptive sensors with onboard maps. Usually a Digital Elevation Map (DEM) is carried onboard the vehicle.

In this context, we consider a radio-altimeter based TAN scenario where transmission restrictions are experienced. For instance, the size and resolution of the uploaded reference map can be constrained when the vehicle is sent to successive areas without a pre-established schedule. Consequently, since we cannot dispose of a high resolution elevation map onboard during the flight, we propose to represent the elevation map by a parametric statistical model, which is plugged into the navigation inference process. More precisely, the altimetric map is modeled using a Gaussian Process (GP) [2]. A GP is completely characterised by a mean function and a covariance kernel. The former can be interpreted as a low resolution approximation of the map, while the latter accounts for the spatial correlations.

Our idea is then to solve the navigation problem by using a Simultaneous Localisation and Mapping (SLAM) [9], [10] approach. More precisely, we jointly estimate a marginalised version of the GP (i.e., some altitudes at fixed x-coordinates), and the position and dynamics of the vehicle. The inference can be carried out by a non-linear Bayesian filter: the Rao-Blackwellised (or marginalised) Particle Filter (RBPF [4], [5]), that takes advantage of the conditional linear Gaussianity of the navigation model. Such a combination of GP and RBPF has been previously proposed for bathymetric SLAM [12], for extended target tracking, [3], and more closely related to our work, for magnetic field SLAM [6]–[8]. However, in [7], [8], the magnetic map is parameterised by a significant number of parameters, and the underlying model is linear as the authors consider a linear GP kernel. Also, since no prior information is exploited, the GP parameters have to be updated over time.

The remainder of the paper is organised as follows. A short introduction on GPs is presented in Section II. Sections III and IV are dedicated to the methodology: in Section III, the GP-based map model for altimetric TAN is discussed, while Section IV introduces our SLAM state space model and the inference algorithm. Section V presents the results in a navigation scenario where the UAV trajectory is planar. Section VI concludes the paper.

II. Gaussian Process Background

A GP [2] is a stochastic process suitable for modelling correlated variables. It can be considered as a Gaussian distribution over functions. The GP is completely characterised by its mean function $\mu(u)$ and covariance kernel $k(u, u')$. Let $f(u)$ be a vector-valued GP, we denote:

$$f(u) \sim \mathcal{GP}\left(\mu(u), k(u, u')\right) \tag{1}$$

where u is an input, and

$$\mu(u) = \mathbb{E}\left[f(u)\right] \tag{2a}$$

$$k(u, u') = \mathbb{E}\left[\left(f(u) - \mu(u)\right)\left(f(u') - \mu(u')\right)^{\top}\right] \tag{2b}$$

A GP is a generalisation of a multivariate Gaussian probability distribution in the sense that the function values evaluated for any finite collection of inputs $\mathbf{u} = [u_1, \ldots, u_N]^\top$ are normally distributed:

$$\begin{bmatrix} f(u_1) \\ \vdots \\ f(u_N) \end{bmatrix} \sim \mathcal{N}(\mu(\mathbf{u}),\, K(\mathbf{u},\mathbf{u})) \tag{3}$$

where $\mu(\mathbf{u}) = [\mu(u_1), \ldots, \mu(u_N)]^\top$ and

$$K(\mathbf{u},\mathbf{v}) = \begin{bmatrix} k(u_1,v_1) & \ldots & k(u_1,v_{Nf}) \\ \vdots & & \vdots \\ k(u_N,v_1) & \ldots & k(u_N,v_{Nf}) \end{bmatrix} \tag{4}$$

with $\mathbf{v} = [v_1, \ldots, v_{Nf}]^\top$, and $\mathcal{N}(m, P)$ denoting the Gaussian distribution of mean m and covariance matrix P.

The GP model can be used to infer an unknown function by exploiting observed data. Let us consider the following measurement model, for $k = 1, \ldots, N$:

$$z_k = f(u_k) + \epsilon_k, \quad \epsilon_k \sim \mathcal{N}(0, R) \tag{5}$$

where z_k is a noisy measurement of the function $f(.)$ for the input u_k, and ϵ_k is an independent and identically distributed (i.i.d.) Gaussian noise. The set of measurements $\mathbf{z} = [z_1, \ldots, z_N]^\top$ can be used to learn the function values $\mathbf{f} = [f(v_1), \ldots, f(v_{Nf})]^\top$ for new inputs $\mathbf{v}$. Their joint distribution is:

$$\begin{bmatrix} \mathbf{z} \\ \mathbf{f} \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} \mu(\mathbf{u}) \\ \mu(\mathbf{v}) \end{bmatrix}, \begin{bmatrix} K(\mathbf{u},\mathbf{u}) + I_N \otimes R & K(\mathbf{u},\mathbf{v}) \\ K(\mathbf{v},\mathbf{u}) & K(\mathbf{v},\mathbf{v}) \end{bmatrix} \right) \tag{6}$$

where $\otimes$ denotes the Kronecker product. From equation (6) one can deduce the conditional distribution (see [2] for more details):

$$\mathbf{f}|\mathbf{z} \sim \mathcal{N}\left(\mu(\mathbf{v}) + A\left(\mathbf{z} - \mu(\mathbf{u})\right), P\right) \tag{7}$$

where

$$A = K(\mathbf{v},\mathbf{u})\left(K(\mathbf{u},\mathbf{u}) + I_N \otimes R\right)^{-1} \tag{8a}$$

$$P = K(\mathbf{v},\mathbf{v}) - A\,K(\mathbf{u},\mathbf{v}) \tag{8b}$$

The equations (7) and (8) are commonly referred to as the GP regression equations [2].

III. Gaussian Process Based Altimetric Map Model for Terrain-Aided Navigation

A. Altimetric terrain-aided navigation

The radar-altimeter measures the distance between the vehicle and the terrain. The UAV is assumed to have a planar motion described in a local frame of reference with axis x and z as represented in Figure 1. Let us denote $p_k^{c,x}$ the x-coordinate and $p_k^{c,z}$ the z-coordinate (altitude) of the vehicle at time k. The measurement model is given by:

$$y_k = p_k^{c,z} - \text{DEM}(p_k^{c,x}) + v_k \tag{9}$$

where v_k is a centered white Gaussian noise of known variance σ_R^2. The DEM is a non-linear function taking as input the position (i.e., the x-coordinate) and yielding the corresponding terrain elevation.

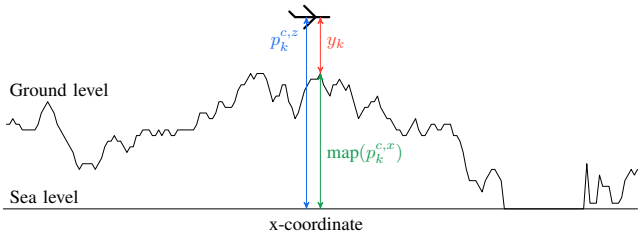

Fig. 1. Measurement model for TAN method.

The TAN method is based on onboard maps of the overflown terrain. When grid-based DEMs are considered, the accuracy of the navigation solution notably depends on the resolution of the latter. For the transmission-limited scenario described in Section I, high resolution DEMs are excluded. Thus, we propose to replace the DEM by a GP statistical model (see Section II).

B. Statistical map model for TAN

As suggested above, we model the reference map by a GP, where only the hyper-parameters of the mean and covariance functions are transmitted. The transmission load depends on the choice of these functions and the associated parameters.

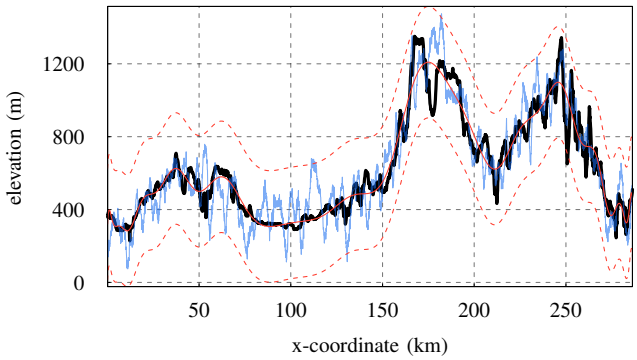

Fig. 2. DEM-based terrain profile (black solid line), and a Gaussian realisation (blue translucent line) with a polynomial mean of 30 degrees (red solid line) and a SE covariance (red dashed lines, 3-σ interval around the mean).

Several options of mean function have been proposed in the GP literature [2] (e.g., polynomial, spline, or deep neural network). The covariance function is generally a squared exponential (SE) kernel with three hyperparameters that can be estimated by Maximum Likelihood (ML) [2], [8]. Figure 2 shows a true terrain profile (black line) and a realisation of a GP-based approximation for a polynomial mean of degree 30 and a SE covariance with three hyperparameters estimated by ML (blue line). A preliminary study on fitting from high resolution DEM data must be carried out in anticipation of the proposed GP-based method, in order to find the mean and covariance functions which contain enough terrain information without being too costly in our reduced transmission scenario.

In the following, we take advantage of this statistical model to perform SLAM. For that purpose, we estimate a marginalised GP at various discretised locations, while inferring the position and motion of the UAV. To build the corresponding state space representation, we use the regression equations (7) and (8). Let us emphasize that the classical TAN method is a special case of this GP-based SLAM model. Indeed, by taking a mean function equal to an approximation of the true terrain (e.g., DEM) and an impulsive covariance kernel (i.e., considering no spatial correlation between the elevation values), we retrieve the classical TAN measurement model (9). Taking into account spatial correlations thanks to the GP formalism makes it possible to develop a SLAM approach that is expected to outperform a coarse map based TAN. In practice, this only requires the storage of three extra parameters, as the kernel is parametric.

IV. SLAM State Space Model and Inference

In this section, we will derive the state space model, consisting of the dynamic equation, the measurement equation and the initial prior density. The interest of the SLAM is to retrieve information on the vehicle position by correlating it with the elevations of chosen reference locations. For that purpose, the vehicle state, denoted $\mathbf{x}_k^c$, is augmented with m unknown elevations given for m fixed x-coordinates denoted $\mathbf{p}_k^{m,x} = \{p_k^{i,x}\}_{i=1:m}$. The elevations are gathered in a vector $\mathbf{h}_k^m = \{h_k^i\}_{i=1:m}$, leading to the following overall state vector, at time k:

$$\mathbf{x}_k = [\mathbf{x}_k^c,\, \mathbf{h}_k^m] \tag{10}$$

with the vehicle state given by:

$$\mathbf{x}_k^c = [p_k^{c,x}, p_k^{c,z}, v_k^{c,x}, v_k^{c,z}]^T \tag{11}$$

where $p_k^{c,x}$ is the x-coordinate (m), $p_k^{c,z}$ the z-coordinate (m), and $v_k^{c,x}$ and $v_k^{c,z}$ are respectively the horizontal and the vertical velocities ($m.s^{-1}$).

A. Motion model

The vehicle dynamic is described by the following quasi-constant velocity model [15]:

$$\mathbf{x}_{k+1}^c = F^c\, \mathbf{x}_k^c + \varepsilon_k^c \tag{12}$$

where $\varepsilon_k^c \sim \mathcal{N}(0, Q^c)$ is i.i.d, and

$$F^c = \begin{pmatrix} I_2 & \Delta_t\, I_2 \\ 0_2 & I_2 \end{pmatrix},\; Q^c = \sigma_q^2 \begin{pmatrix} \Delta_t^3/3\, I_2 & \Delta_t^2/2\, I_2 \\ \Delta_t^2/2\, I_2 & \Delta_t\, I_2 \end{pmatrix} \tag{13}$$

In equation 13, σ_q stands for the standard deviation of the acceleration.

In the present work, for the sake of simplicity, we choose to keep the m x-coordinates unchanged over time $\mathbf{p}_k^{m,x}$. Thus, the dynamic of the m unknown elevations is constant:

$$\mathbf{h}_{k+1}^m = \mathbf{h}_k^m \tag{14}$$

Finally, we build an augmented description of the dynamics together with the initial a priori density:

$$\mathbf{x}_{k+1} = F\, \mathbf{x}_k + \varepsilon_k, \quad \varepsilon_k \sim \mathcal{N}(0,\, Q) \tag{15a}$$

$$\mathbf{x}_0 = \mathcal{N}(\mathbf{m}_0,\, P_0) \tag{15b}$$

where

$$F = \begin{pmatrix} F^c & 0_m \\ 0_m & I_m \end{pmatrix}, \quad Q = \begin{pmatrix} Q^c & 0_{2\times m} \\ 0_{m\times 2} & 0_{m\times m} \end{pmatrix} \tag{16a}$$

$$\mathbf{m}_0 = \begin{bmatrix} \mathbf{m}_0^c \\ \mu\,(\mathbf{p}_0^{m,x}) \end{bmatrix}, \quad P_0 = \begin{pmatrix} P_0^c & 0_{2\times m} \\ 0_{m\times 2} & K\,(\mathbf{p}_0^{m,x}, \mathbf{p}_0^{m,x}) \end{pmatrix} \tag{16b}$$

where μ and K are respectively the mean vector and covariance matrix of the prior marginalised GP.

B. Measurement model

The measurement equation expresses the relationship between the vehicle state and the elevations. It can be obtained by using the GP regression equations (7) and (8) (see Section III). It takes the form:

$$y_k = p_k^{c,z} - [H\,(\mathbf{h}_k^m - \mu\,(\mathbf{p}_k^{m,x})) + \mu\,(p_k^{c,x})] + v_k' \tag{17}$$

where $v_k' \sim \mathcal{N}(0,\, \sigma_R^2 + \sigma_{R^m}^2)$, and

$$H = K_k^{c,m}\,(K_k^{m,m})^{-1} \tag{18a}$$

$$\sigma_{R^m}^2 = k(p_k^{c,x}, p_k^{c,x}) - K_k^{c,m}\,(K_k^{m,m})^{-1}\,(K_k^{c,m})^T \tag{18b}$$

with $K_k^{c,m} \triangleq K(p_k^{c,x}, \mathbf{p}_k^{m,x})$, and $K_k^{m,m} \triangleq K(\mathbf{p}_k^{m,x}, \mathbf{p}_k^{m,x})$. Due to the GP, the noise should be correlated over time, but, based on the same assumption as in [3], we don't take into account this dependency.

C. Inference algorithm

From the state space model (15)-(17) derived in the previous sections, it is possible to compute the posterior distribution of the state by using standard inference techniques. In this work, the inference is carried out by the RBPF [4], [5]. The RBPF is a variance reduction method for conditionally linear Gaussian models. The principle is to separate the state variables into two groups. The non-linear part is dealt with a particle filter, thereafter a Regularised Particle Filter [14] (RPF), while a Kalman filter is applied to the conditionally linear part. In this work, the non-linear part is the vehicle state (position and velocity) and the conditionally linear part is composed of the elevations.

The RPF is based on kernel estimation approaches, which provides more accuracy by considering mixtures of weighted bounded kernels. The kernel and its associated bandwidth parameter are chosen to minimise the Mean Integrated Square Error (MISE) between the true conditional density and the corresponding regularised empirical density. When all the particles have the same weight, which is the case immediately after the resampling step, the optimal kernel is the Epanechnikov kernel [14].

As for the implementation, the theoretical load in terms of total number of floating point operations for the RBPF is provided in [16] assuming linear dynamics. The costs introduced by the proposed measurement equation (17) are limited, as the matrix $K_k^{m,m}$ remains the same over time and over the particles, and therefore is inverted only once. Since the complexity is polynomial with respect to the number of

particles, the computation can be performed in real-time on Graphics Processing Unit. Furthermore, particle filter operations are highly parallelisable in practice [17], making them suitable for real-time applications.

V. Application

The reference map and the generation of the UAV trajectory are described in Section V-A. The choice of the GP mean and covariance functions are discussed in Section V-B. The simulation parameters are summarised in Section V-C. Finally, the results are presented in Section V-D.

A. Reference trajectory and map

The reference trajectory of the UAV is simulated using the state model (12). Thus, as illustrated by the red solid line in Figure 3, the vehicle follows a quasi uniform rectilinear motion. The reference DEM [13] is extracted from the Shuttle Radar Topography Mission (SRTM), and is represented by the black solid line in Figure 3. The vehicle flies over a steep terrain profile with significant elevation changes and ambiguities, which is expected to make the navigation task difficult.

B. GP-based measurement models

The measurement model is detailed in Section IV-B. The mean function considered for the GP-based SLAM method is a polynomial function of degree 30, as illustrated in Figure 2. The mean parameters are estimated by a least-squares fit. The chosen GP covariance function is the Squared Exponentiel (SE) kernel, which is parameterised by three hyperparameters: the length-scale l_c, the signal variance σ_f^2, and the noise variance σ^2. The SE kernel expression is:

$$K_\theta(u,v) = \sigma_f^2 \exp\left(-\frac{1}{2\,l_c^2}|u-v|^2\right) + \sigma^2\,\delta_{uv} \qquad (19)$$

where $\theta = [l_c, \sigma_f, \sigma]$, and δ_{uv} is the Kronecker delta function, i.e., $\delta_{uv} = 1$ if $u = v$, and 0 otherwise. The hyperparameters of the GP covariance function are estimated by ML and are approximately equal to $\theta \approx [1000, 100, 10]$.

C. Parameters

The rest of the parameters is given in Table I. For the RPF inside the RBPF, the regularisation bandwidth parameter of the Epanechnikov kernel [14] is also provided in Table I.

TABLE I
Simulation settings.

Sampling period	$\Delta_t = 6$ s
Number of altimetric measurements	700
Number of elevations to estimate	$m = 100$
Number of particles	$N = 2000$
Resampling threshold	$N_{th} = 0.75\ N$
Regularisation bandwidth parameter	$\mu = 0.3$
Initial position	$[120000, 5000]^\top\ m$
Initial velocity	$[100, 0]^\top\ km/h$
Initial uncertainty in position (st.d.)	$[500, 5]\ m$
Initial uncertainty in velocity (st.d.)	$[0.5, 0.05]\ m/s$
Process noise (st.d.)	$\sigma_q = 0.01\ m/s$
Measurement error (st.d.)	$\sigma_R = 5\ m$

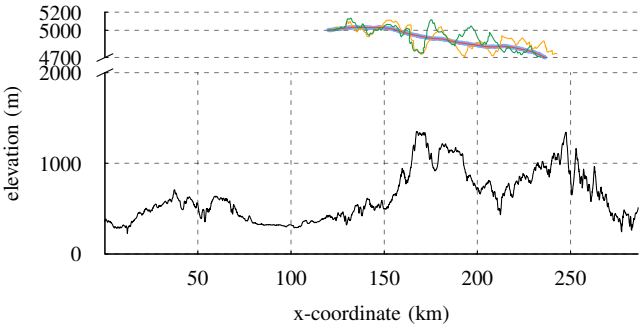

Fig. 3. True vehicle trajectory (red solid line), and three estimated trajectories: the one estimated by a RPF (blue translucent line), a RBPF (green solid line), and a RBPF trajectory averaged over 35 Monte Carlo runs (orange solid line).

D. Results: a comparison between a standard TAN and our SLAM approach

In order to evaluate the performance of the proposed SLAM approach, we compare it to a standard TAN formulation. The TAN method is implemented using an RPF [14], where only the vehicle state vector $\mathbf{x}_k^c$ is estimated. In this scenario, a high-resolution DEM is carried onboard, thus the RPF yields very accurate estimates as illustrated by the blue translucent line that follows the true trajectory (red solid line) in Figure 3.

To achieve this ideal navigation, the UAV must receive a DEM of 3181 map points (x and z coordinates). In the proposed SLAM approach, a trade-off is made between the amount of data to be transferred and the accuracy of the navigation. For our choice of GP mean and covariance functions, it only requires the transmission of 33 parameters. As a counterpart, the estimation of the vehicle state is slightly degraded, which represents for the z-coordinate a mean error of $60\,m$ for 35 Monte Carlo runs. Nevertheless, the trajectory obtained by the RBPF, e.g., the green solid line, still follows the reference trajectory represented by the red solid line in Figure 3. It should be noted that the RBPF exhibits a stable behavior in the sense that, whatever the noise realisations, the estimated trajectory remains close to the true one. It is confirmed by the trajectory averaged over 35 Monte Carlo runs, and depicted in the orange solid line.

Besides providing a satisfying navigation solution, the proposed SLAM method uses the spatial correlations captured by the GP to refine the estimation of some elevations. For a Monte Carlo run, the estimation of the elevations, at three different steps of the UAV trajectory, is shown in Figure 4. The spatial correlation captured by our GP-based map approximation allows us to estimate with precision only the elevations underneath the UAV trajectory depending on the SE kernel length-scale. Indeed, the 3-σ interval narrows around the x-estimate of the UAV as it navigates.

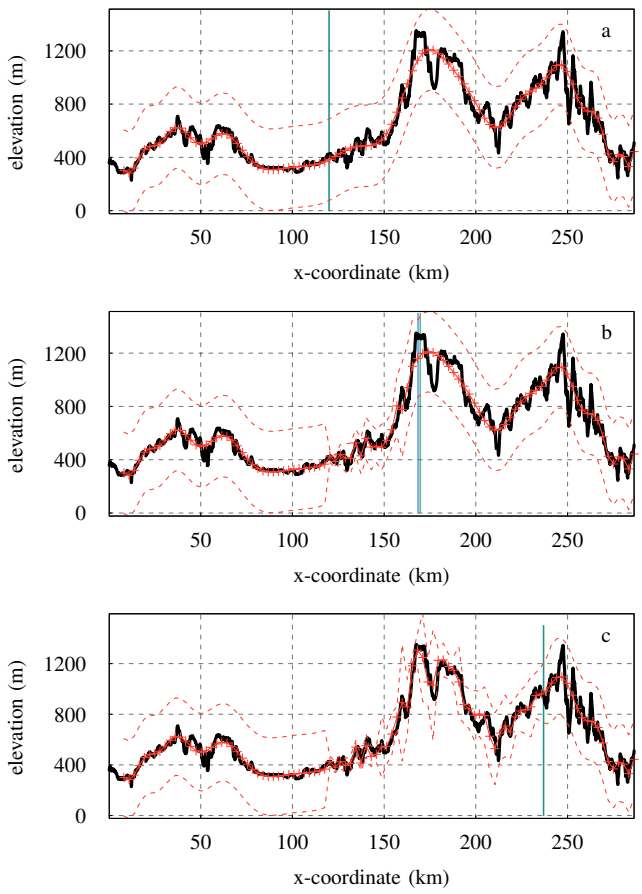

Fig. 4. DEM-based reference terrain profile (black solid line), and the hundred estimated elevations (red solid line) at the x-coordinates represented by red crosses, with the associated 3-σ interval (red dashed lines). Three different instants of the trajectory are represented: the initial time before the first measurement (figure a), at the middle of the trajectory (after 350 altimetric measurements - figure b), and at the last measurement (figure c). The true position of the vehicle is represented by the blue solid vertical line and the RBPF estimated position is given by the green solid vertical line.

As the UAV trajectory is not a priori known, a significant number of elevations (here 100) must be estimated to cover the entire terrain profile. However, not all the elevations can be refined, as illustrated by the constant 3-σ interval at the left and right of the trajectory. In order to reduce the number of elevations and keep only relevant ones, the m x-coordinates could be dynamically moved according to the UAV estimated position.

VI. Conclusion and Perspectives

In this paper, we have proposed a solution for performing navigation with a limited-size reference map (summarised by only 33 parameters instead of the 3000 map points). The originality is to represent the map by a Gaussian process that is well-suited to model spatial correlations in the elevation profile. The inference is led through a Rao-Blackwellised particle filter. The results on a 2D radio-altimetric scenario demonstrate its ability to estimate the trajectory while refining the map.

Several perspectives can be mentioned. A possible direction of future work could be to dynamically move the x-coordinates of the unknown elevations according to the estimated position of the vehicle, which could drastically accelerate the SLAM process. Another direction would consist in extending the approach to 3D scenarios. In such cases, the predetermination of the GP map will be more challenging, involving non-stationarity or piecewise non-stationarity.

References

[1] G. Balamurugan, J. Valarmathi, and V. Naidu, "Survey on UAV navigation in GPS denied environments," in 2016 International conference on signal processing, communication, power and embedded system (SCOPES), pp. 198–204, 2016.

[2] C.E. Rasmussen, and C.K.I. Williams, "Gaussian processes for machine learning," in Springer, 2006.

[3] N. Wahlström, and E. Özkan, "Extended target tracking using Gaussian processes," in IEEE Transactions on Signal Processing, v. 63, n. 16, pp. 4165–4178, 2015.

[4] G. Casella, and C.P. Robert, "Rao-Blackwellisation of sampling schemes," in Biometrika, Oxford University Press, v. 83, n.1, pp. 81–94, 1996.

[5] P.-J. Nordlund, "Sequential Monte Carlo filters and integrated navigation," in Citeseer, 2002.

[6] A. Solin, M. Kok, N. Wahlström, T.B. Schön, and S. Särkkä, "Modeling and interpolation of the ambient magnetic field by Gaussian processes," in IEEE Transactions on robotics, v. 34, n. 4, pp. 1112–1127, 2018.

[7] F. Viset, R. Helmons, and M. Kok, "An Extended Kalman Filter for Magnetic Field SLAM Using Gaussian Process Regression," in Sensors, MDPI, v. 22, n. 8, pp. 2833, 2022.

[8] M. Kok, and A. Solin, "Scalable magnetic field SLAM in 3D using Gaussian process maps," in IEEE 21st international conference on information fusion (FUSION), pp. 1353–1360, 2018.

[9] S. Thrun, D. Fox, W. Burgrard, and F. Dellaert, "Robust Monte Carlo localization for mobile robots," in Artificial intelligence, Elsevier, v. 128, n. 1-2, pp. 99–141, 2001.

[10] S. Thrun, "Simultaneous localization and mapping," in Robotics and cognitive approaches to spatial mapping, Springer, pp. 13–41, 2008.

[11] S. Vasudevan, F. Ramos, E. Nettleton, and H. Durrant-Whyte, "Gaussian process modeling of large-scale terrain," in Journal of Field Robotics, Wiley Online Library, v. 26, n. 10, pp. 812–840, 2009.

[12] S. Barkby, S.B. Williams, O. Pizarro, and M.V. Jakuba, "Bathymetric particle filter SLAM using trajectory maps," in The International Journal of Robotics Research, SAGE Publications Sage UK: London, England, v. 31, n. 12, pp. 1409–1430, 2012.

[13] A. Jarvis, H.I. Reuter, A. Nelson, and E. Guevara, "Hole-filled seamless SRTM data V4," International Centre for Tropical Agriculture (CIAT), , available from http://srtm.csi.cgiar.org, 2008.

[14] C. Musso, N. Oudjane, and F. Le Gland, "Improving regularised particle filters," in Sequential Monte Carlo methods in practice, Springer, pp. 247–271, 2001.

[15] Y. Bar-Shalom, X.R. Li, and T. Kirubarajan, "Estimation with applications to tracking and navigation: theory algorithms and software," in John Wiley & Sons, 2001.

[16] R. Karlsson, T. Schon, and F. Gustafsson, "Complexity analysis of the marginalized particle filter," in IEEE Transactions on Signal Processing, v. 53, n. 11, pp. 4408–4411, 2005.

[17] M. Bolic, "Architectures for efficient implementation of particle filters," in State University of New York at Stony Brook, 2004.