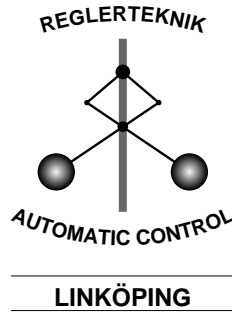# Terrain Navigation using Bayesian Statistics

Niclas Bergman, Lennart Ljung, and Fredrik Gustafsson

Department of Electrical Engineering
Linköping University, S-581 83 Linköping, Sweden
WWW: http://www.control.isy.liu.se
Email: ljung@isy.liu.se, fredrik@isy.liu.se

1999-03-11

REGLERTEKNIK

AUTOMATIC CONTROL

LINKÖPING

Terrain Navigation using Bayesian Statistics
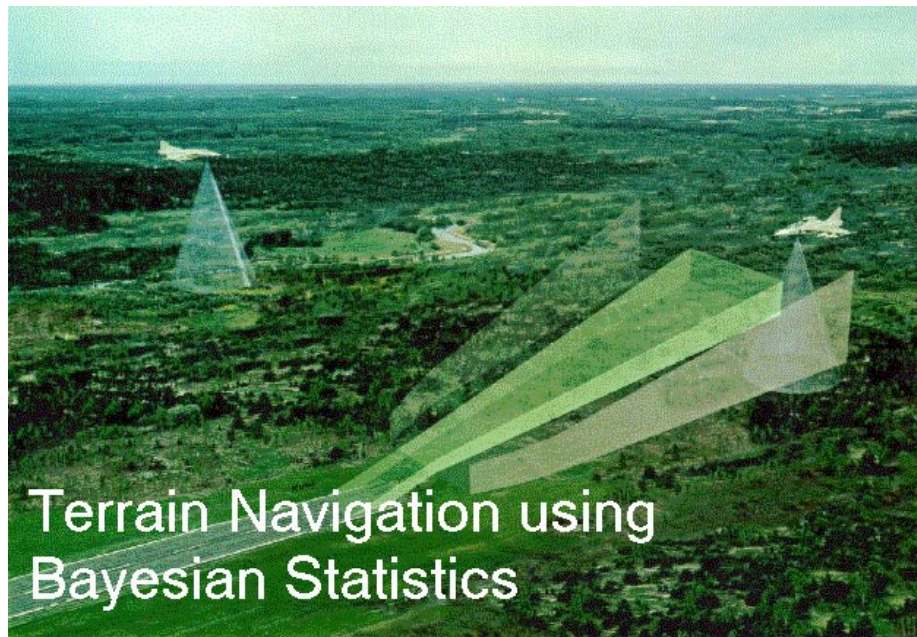
Niclas Bergman, Lennart Ljung and Fredrik Gustafsson

Dept. of Electrical Engineering, Linköping University, Sweden

E-mail: {niclas,ljung,fredrik}@isy.liu.se

URL: http://www.control.isy.liu.se

In aircraft navigation the demands on reliability and safety are very high. The importance of accurate position and velocity information becomes crucial when flying an aircraft at low altitudes, and especially during the landing phase. Not only should the navigation system have a consistent description of the position of the aircraft, but also a description of the surrounding terrain, buildings and other objects that are close to the aircraft. Terrain navigation is a navigation scheme that utilizes variations in the terrain height along the aircraft flight path. Integrated with an Inertial Navigation System (INS), it yields high performance position estimates in an autonomous manner, i.e., without any support informa-

tion sent to the aircraft. In order to obtain these position estimates, a nonlinear recursive estimation problem must be solved on-line. Traditionally, this filtering problem has been solved by local linearization of the terrain at one or several assumed aircraft positions. Due to changing terrain characteristics, these linearizations will in some cases result in diverging position estimates. In this work, we show how the Bayesian approach gives a comprehensive framework for solving the recursive estimation problem in terrain navigation. Instead of approximating the model of the estimation problem, the analytical solution is approximately implemented. The proposed navigation filter computes a probability mass distribution of the aircraft position and updates this description recursively with each new measurement. The navigation filter is evaluated over a commercial terrain database, yielding accurate position estimates over several types of terrain characteristics. Moreover, in a Monte Carlo analysis, it shows optimal performance as it reaches the Cramér-Rao lower bound.

1   Aircraft Navigation

Navigation is the concept of determination of the kinematic state of a moving vehicle. In aircraft navigation this usually consists of finding the position and velocity of the aircraft. Accurate knowledge of this state is critical for flight safety. Therefore, an aircraft navigation system should not only provide a reliable and accurate estimate of the current kinematic state of the aircraft, but also a consistent description of the accuracy of this estimate.

Aircraft navigation is typically performed using a combination of dead-reckoning and fix position updates. In dead-reckoning systems, the state vector is calculated from a continuous series of measurements of the aircraft movement relative to an initial position. Due to error accumulation, dead-reckoning systems must be re-initialized periodically. Fix point, or positioning, systems measure the state vector more or less without regard to the previous movement of the aircraft. They are therefore suitable for re-initialization of dead-reckoning systems.

The most common dead-reckoning systems are the Inertial Navigation Systems (INS) in which accelerometers are used to sense the magnitude of the

aircraft acceleration. A set of gyroscopes either maintains the accelerometers in a known orientation with respect to a fixed, non-rotating coordinate system, commonly referred to as inertial space, or measures the angular rate of the accelerometers relative to inertial space. The inertial navigation computer uses these sensed accelerations and angular rates to compute the aircraft velocity, position, attitude, attitude rate, heading, altitude, and possibly range and bearing to destination. An INS generates near instantaneous continuous position and velocity, it is self-contained, functions at all latitudes, and in all weather conditions. It operates independently of aircraft manoeuvres and without the need for ground station support. Complete and comprehensive presentations of inertial navigation can be found in [14, 16].

Positioning systems that have attracted a lot of attention lately are the global satellite navigation systems which promise a very high accuracy and global coverage. There are two global satellite systems for navigation in use today: GPS, developed by the U.S., and the Russian system GLONASS. Position estimates are obtained by comparing distances from the aircraft to four or more satellites. The systems have been developed for military purposes and several coding techniques are used to keep the accuracy for civilian or unauthorized users at a level far from the actual performance of the systems. However, using ground stations as reference, the coding errors can be removed efficiently. Vendors have off-the-shelf receivers for differential GPS (DGPS) with a position accuracy below the one meter level. A comprehensive summary of the concept of satellite navigation can be found in [14, Chapter 5].

The radio navigation systems have the disadvantage of relying on information broadcasted to the aircraft. This information could be deliberately jammed in a hostile situation, or the transmitters could be destroyed, leaving the aircraft without navigation support. Hence, even if the satellite systems give high accuracy position information they need to be combined with alternative backup systems using other navigation principles. The concept of terrain navigation is an alternative positioning technique that autonomously generates updates to the INS, although in general not with the same accuracy as the satellite systems. The main idea in terrain navigation is to measure the variations in the terrain height underneath the aircraft flight path and compare these measurements

with a reference map. The principle of terrain navigation is depicted in Fig-
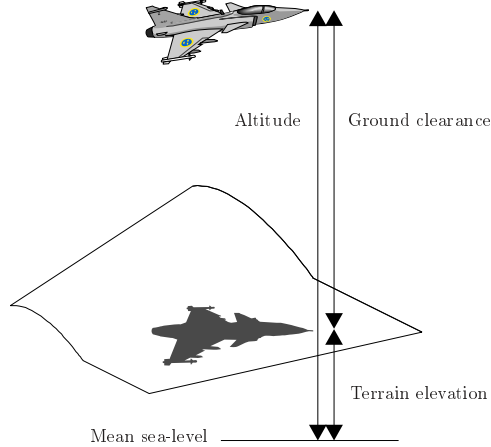

Figure 1: The principle of terrain navigation.

ure 1. The aircraft altitude over mean sea-level is measured with a barometric altimeter and the ground clearance is measured with a radar altimeter, pointing downward. The terrain elevation beneath the aircraft is found by taking the difference between the altitude and ground clearance measurements. The navigation computer holds a digital reference map with values of the terrain elevation as a function of longitude and latitude. The measured terrain elevation is compared with this reference map and matching positions in the map are determined. Terrain repetitiveness and flatness make this matching nontrivial and the quality of the outcome dependent on the amount of terrain variation. Many areas inside the reference map will in general have a terrain elevation comparable to the measured one. In order to distinguish the true position from false ones, several measurements along the aircraft flight path need to be considered. Hence, the measurements must be matched with the map on-line and in a recursive manner. For comprehensive discussions about the applications of terrain navigation techniques, see [10, 11].

The performance of the matching of terrain elevation measurements with the map depends highly on the type of terrain in the area. Flat terrain gives little or no information about the aircraft position. Rough, but repetitive, terrain can

give several well matched positions in an area, making it hard to distinguish between several, well matching tracks. The information content inside a generic area of the map can be shown to be proportional to the average size of the terrain gradient,

$$\sqrt{\frac{1}{N}\sum_{i=1}^{N}\|\nabla h(x_i)\|^2} \qquad (1)$$

where x_i are positions uniformly distributed in the area of interest. This scalar measure of the terrain information can be connected to an associated Cramér-Rao bound for the underlying estimation problem [3]. The right part of Figure 2 shows (1) evaluated in square blocks of 400 meter side where bright color indicates a large value. The terrain map used in this work is a real commercial map
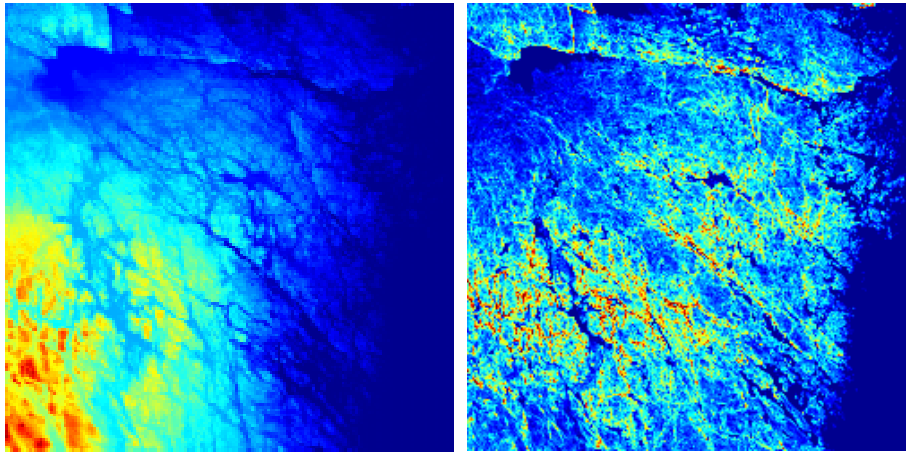

Figure 2: The left part shows the terrain height and the right part the information content in the map over a central part of Sweden.

of a 100 by 100 km area of central Sweden. The pure terrain elevation samples are given in a uniform mesh of 50 by 50 meter resolution and shown to the left in the figure. Using interpolation from surrounding map values, the terrain map can be regarded as a known look-up table of terrain elevations as a function of position. Everything that in the real world cannot be found by interpolation from the database values must be regarded as noise. The right part of Figure 2 very clearly shows the lakes, and the coastline of the Baltic sea to the right in

the map. The north-west part of the map is flat agricultural land with very little navigation information, while the southern part consists of very rough terrain with varying hills of some hundred meters height and narrow valleys that give a high information content. Maps such as the right part of Figure 2 could be used for mission planning purposes, e.g., finding the most informative path to the final destination.

There are several commercial algorithms that solve the terrain navigation problem. Since the development has been driven by military interests, most of these are not very well documented in the literature. The most frequently referred algorithms for terrain navigation are TERCOM (terrain contour matching) and SITAN (Sandia inertial terrain-aided navigation). TERCOM is batch oriented and correlates gathered terrain elevation profiles with the map periodically [2, 9, 16]. The aircraft is not allowed to maneuver during data acquisition in TERCOM and therefore it has mainly been used for autonomous crafts, like cruise missiles. SITAN is recursive and uses a modified version of an extended Kalman filter (EKF) in its original formulation [13]. When flying over fairly flat or over very rough terrain, or when the aircraft is highly maneuverable, this algorithm does not in general perform well. In order to overcome these divergence problems parallel EKFs have been used in [6, 12]. Another widespread system is the TERPROM system, developed by British Aerospace, can be found in several NATO aircraft. It is a hybrid solution, in which an acquisition-mode correlates measurements in batch to find an initial position and in track-mode processes measurements recursively using Kalman filter techniques. However, due to commercial interests and because of its use as a classified military system, it is not as well documented in the literature as the previous two. One more recent and different approach that tries to deal with the nonlinear problems is VATAN [8]. In VATAN the Viterbi algorithm is applied to the terrain navigation problem, yielding a maximum a posteriori position estimate.

In this work, we take a completely statistical view on the problem and solve the matching with the map as a recursive nonlinear estimation problem. The conceptual solution is described in the following section and an approximate implementation in Section 3. Simulation results with this implementation are presented Section 4.

2 The Bayesian Approach to Terrain Navigation

As shown in Figure 1 the difference between the altitude estimate and the measured ground clearance yields a measurement of the terrain elevation. Assuming additive measurement noise the terrain elevation y_t relates to the current aircraft position x_t according to

$$y_t = h(x_t) + e_t \tag{2}$$

where the function $h(\cdot) : \mathbb{R}^2 \mapsto \mathbb{R}$ is the terrain elevation map. The measurement noise e_t is a white process with some known distribution $p_{e_t}(\cdot)$. This measurement error models both the errors in the radar altimeter measurements, the current altitude estimate and errors originating from the interpolation in the terrain map not perfectly resembling the real world. Let u_t denote the estimate of the relative movement of the aircraft between two measurements obtained from the INS. Modeling the dead-reckoning drift of the INS with a white additive process v_t, the absolute movement of the aircraft obeys a simple linear relation

$$x_{t+1} = x_t + u_t + v_t \tag{3}$$

where v_t is distributed according to some assumed known probability density function $p_{v_t}(\cdot)$. Summarizing equations (2) and (3) yields the nonlinear model

$$\begin{aligned} x_{t+1} &= x_t + u_t + v_t \\ y_t &= h(x_t) + e_t \end{aligned} \qquad t = 0, 1 \ldots \tag{4}$$

where v_t and e_t are mutually independent white processes, both of them uncorrelated with the initial state x_0 which is distributed according to $p(x_0)$. One may argue that the INS estimate u_t should be regarded as a sensor measurement instead of as a known parameter in the state transition equation,

$$u_t = x_{t+1} - x_t + e_t^{\text{INS}}.$$

This would require the introduction of a new state vector incorporating both x_t and x_{t+1} in order to retain the Markovian property of the state space model. Instead, we choose to include the error e_t^{INS} in the process noise v_t in (4).

This limits the state dimension and drastically reduces the computational power required to recursively compute an approximation of the conditional density of the states.

The objective of the terrain navigation algorithm is to estimate the current aircraft position x_t using the observations collected until present time

$$\mathbb{Y}_t = \{y_i\}_{i=0}^t.$$

With a Bayesian approach to recursive filtering, everything worth knowing about the state at time t is condensed in the conditional density $p(x_t \mid \mathbb{Y}_t)$. With some abuse of notation, the distribution of a generic random variable z conditioned on another related random variable w is

$$p(z \mid w) = \frac{p(z, w)}{p(w)} = \frac{p(w \mid z) p(z)}{p(w)} \tag{5}$$

Assume that $p(x_t \mid \mathbb{Y}_{t-1})$ is known and apply (5) to the last member in the set $\mathbb{Y}_t$,

$$p(x_t \mid \mathbb{Y}_t) = \frac{p(y_t \mid x_t, \mathbb{Y}_{t-1}) p(x_t \mid \mathbb{Y}_{t-1})}{p(y_t \mid \mathbb{Y}_{t-1})}.$$

Inserting the model (4) and noting that the denominator is a scalar normalization constant yield,

$$p(x_t \mid \mathbb{Y}_t) = \alpha_t^{-1} p_{e_t}(y_t - h(x_t)) p(x_t \mid \mathbb{Y}_{t-1})$$
$$\alpha_t = \int p_{e_t}(y_t - h(x_t)) p(x_t \mid \mathbb{Y}_{t-1}) dx_t$$

which describes the influence of the measurement. Using (5) the joint density of the states at two measurement instants is

$$p(x_{t+1}, x_t) = p(x_{t+1} \mid x_t) p(x_t).$$

The density update between two measurements is found by marginalizing this expression on the state x_t and inserting (4),

$$p(x_{t+1} \mid \mathbb{Y}_t) = \int p_{v_t}(x_{t+1} - x_t - u_t) p(x_t \mid \mathbb{Y}_t) dx_t.$$

This completes one iteration of the recursive solution. Summarizing the derivation, the Bayesian formula for updating the conditional density is initiated by

$p(x_0 \mid \mathbb{Y}_{-1}) = p(x_0)$ and calculated as

$$
\boxed{
\begin{aligned}
p(x_t \mid \mathbb{Y}_t) &= \alpha_t^{-1}\, p_{e_t}(y_t - h(x_t))\, p(x_t \mid \mathbb{Y}_{t-1}) \\
p(x_{t+1} \mid \mathbb{Y}_t) &= \int p_{v_t}(x_{t+1} - x_t - u_t) p(x_t \mid \mathbb{Y}_t)\, dx_t
\end{aligned}
}
\tag{6}
$$

where

$$
\alpha_t = \int p_{e_t}(y_t - h(x_t)) p(x_t \mid \mathbb{Y}_{t-1})\, dx_t.
$$

The Bayesian solution is a density function describing the distribution of the states given the collected measurements. From the conditional density, a point estimate such as the minimum mean square error estimate can be formed

$$
\hat{x}_t = \int x_t\, p(x_t \mid \mathbb{Y}_t)\, dx_t.
\tag{7}
$$

Assuming that this estimate is unbiased, the covariance

$$
C_t = \int (x_t - \hat{x}_t)(x_t - \hat{x}_t)^T\, p(x_t \mid \mathbb{Y}_t)\, dx_t
\tag{8}
$$

quantizes the accuracy of the estimate. Equation (8) is convenient when comparing (7) with estimates from other navigation systems.

The recursive update of the conditional density (6) describes how the measurement y_t and the relative movement u_t affect the knowledge about the aircraft position. With each new terrain elevation measurement, the prior distribution $p(x_t \mid \mathbb{Y}_{t-1})$ is multiplicatively amplified by the likelihood of the measurement y_t. This means that the conditional probability will decrease in unlikely areas and increase in areas where it is likely that the measurement was obtained. Between two measurements, the density function $p(x_t \mid \mathbb{Y}_t)$ is translated according to the relative movement of the aircraft obtained from the INS and convolved with the density function of the error of this estimate. Thus the support and shape of the conditional density will adapt to areas which fit the measurements well and follow the movement obtained from the INS. It is worth noting that (6) is the Bayesian solution to (4) for all possible nonlinear functions $h(\cdot)$ and for any noise distributions $p_{v_t}(\cdot)$ and $p_{e_t}(\cdot)$. In the special case of linear measurement equation and Gaussian distributed noises the equations above coincide with the Kalman filter [1].

9

Computationally, each iteration of the Bayesian solution (6) consists of solving several integrals. Due to the unstructured nonlinearity $h(\cdot)$, these integrations are in general impossible to solve in closed form and therefore there exists no solution that updates the conditional density analytically. The implementation must therefore inevitably be approximate. A straightforward way to implement the solution is to simply evaluate the recursion in several positions inside the area where the aircraft is assumed to be and update these values further through the recursion. With such a quantization of the state space, the integrals in (6) turn into sums over the chosen point values. The earliest reference of such a numerical approach to solving the nonlinear filtering problem is [7]. More recent references involve the p-vector approach in [17] and a slightly different approach, presented in [15], using a piecewise constant approximation to the density function. In the terrain navigation problem the state dimension is two and the quantization can in general be viewed as a bed-of-nails where the length of each nail corresponds to a certain elementary mass in that position. The implementation described in this paper is therefore labeled the point-mass filter (PMF).

3 The Point-Mass Filter

Assume that N grid points in $\mathbb{R}^2$ have been chosen for the approximation of $p(x_t \mid \mathbb{Y}_t)$. Introduce the notation

$$x_t(k) \qquad k = 1, 2, \ldots, N$$

for these N vectors in $\mathbb{R}^2$. Each of these N grid points has a corresponding probability mass

$$p(x_t(k) \mid \mathbb{Y}_t) \qquad k = 1, 2, \ldots, N.$$

In order to obtain a simple and efficient algorithm, the grid points are chosen from a uniform rectangular mesh with resolution of δ meters between each grid point. Each integral operation in (6) is approximated by a finite sum over the

grid points with nonzero weight

$$\int_{\mathbb{R}^2} f(x_t)\, dx_t \approx \sum_{k=1}^{N} f(x_t(k))\, \delta^2.$$

Applying this approximation to (6) yields the Bayesian point-mass recursion:

$$
\begin{aligned}
p(x_t(k) \mid \mathbb{Y}_t) &= \alpha_t^{-1} p_{e_t}(y_t - h(x_t(k)))\, p(x_t(k) \mid \mathbb{Y}_{t-1}) \\
x_{t+1}(k) &= x_t(k) + u_t \quad k = 1, 2, \ldots, N \\
p(x_{t+1}(k) \mid \mathbb{Y}_t) &= \sum_{n=1}^{N} p_{v_t}(x_{t+1}(k) - x_t(n))\, p(x_t(n) \mid \mathbb{Y}_t)\, \delta^2
\end{aligned}
\tag{9}
$$

where

$$\alpha_t = \sum_{k=1}^{N} p_{e_t}(y_t - h(x_t(k)))\, p(x_t(k) \mid \mathbb{Y}_{t-1})\, \delta^2. \tag{10}$$

The time update has been split into two parts. First the grid points are translated with the INS relative movement estimate u_t and then the probability mass density is convolved with the density $p_{v_t}(\cdot)$. The point estimate (7) is computed at each iteration as the center of mass of the point-mass density,

$$\hat{x}_t = \sum_{k=1}^{N} x_t(k) p(x_t(k) \mid \mathbb{Y}_t)\, \delta^2.$$

Hence, the estimate does not necessarily fall on a grid point.

In order to follow the aircraft movements the grid must be adapted to the support of the conditional density. After each measurement update, every grid point with a weight less than $\varepsilon > 0$ times the average mass value

$$\frac{1}{N} \sum_{k=1}^{N} p(x_t(k) \mid \mathbb{Y}_t) = \frac{1}{N\delta^2}.$$

is removed from the grid. The new set of grid points is defined by

$$\left\{ x_t(k) : p(x_t(k) \mid \mathbb{Y}_t) > \varepsilon / N\delta^2 \right\}.$$

The weights need to be re-normalized after this truncation operation. The truncation will make the algorithm focus on areas with high probability and remove grid points in areas where the conditional density is small. The basic grid resolution δ will however not be affected by the truncation. When the algorithm

11

is initialized, the uncertainty about the aircraft position is usually rather high. The prior will then have a large support, and naturally it is not interesting to have a high grid resolution. Instead we start with a sparse grid and run the algorithm and remove weights using the truncation operation above until the number of remaining grid points falls below some threshold N_0. Then the mesh resolution can be increased and the algorithm continued to process new measurements, updating the conditional density in the new dense grid. The up-sampling is performed by placing one grid point between every neighboring grid point in the mesh using linear interpolation to determine its weight. This will yield a doubling of the mesh resolution. The convolution in (9) will introduce some extra grid points along the border of the point-mass approximation, increasing the support of the mesh. If the measurements have low information content there will be a net-increase of grid points even though some are removed by the truncation operation. Therefore, if the number of grid points increases above some threshold N_1, the mesh is decimated by removing every second grid point from the mesh, halving the mesh resolution.

Hence, the number of grid points N, the point-mass support and the mesh resolution δ is automatically adjusted through each iteration of the algorithm using the design parameters ε, N_0 and N_1. An illustration is given in Figure 6. In summary the PMF algorithm consists of (9) and the appropriate resampling of the grid described above. Details about the implementation and the grid refinement procedure can be found in [3].

The refinement of the grid support and resolution in the PMF described above is of course *ad hoc*, and one may wonder if this actually works. The Cramér-Rao lower bound is a fundamental limit on the achievable algorithm performance which can be used to evaluate the average performance of the PMF and verify that the filter solves the nonlinear filtering problem with near optimal performance. The results are presented here without proofs or derivations, see [3–5] for details. Let $N(m, P)$ denote the n-dimensional Gaussian distribution with mean vector μ and covariance matrix P

$$N(\mu, P) = \frac{1}{\sqrt{(2\pi)^n |P|}} \exp\left(-\tfrac{1}{2}(x - \mu)^T P^{-1}(x - \mu)\right).$$

Inserting Gaussian distributions in (4),

$$p_{e_t}(\cdot) = \mathrm{N}(0, R_t) \qquad p_{v_t}(\cdot) = \mathrm{N}(0, Q_t) \qquad p(x_0) = \mathrm{N}(\hat{x}_0, P_0),$$

the Cramér-Rao lower bound for the one step ahead prediction of the states satisfies the matrix (Riccati) recursion,

$$P_{t+1} = P_t - P_t H_t (H_t^T P_t H_t + R_t)^{-1} H_t^T P_t + Q_t$$

initiated with P_0. Above H_t is the gradient of $h(\cdot)$ evaluated at the true state value at time t,

$$H_t = \nabla h(x_t).$$

Thus, the Cramér-Rao bound is a function of the noise levels and the gradient of the terrain along the true state sequence.

The Cramér-Rao bound sets a lower limit on the estimation error covariance which depends on the statistical properties of the model (4) and on the algorithm used. A Monte Carlo simulation study is performed to determine the average performance of the algorithm for comparison with the Cramér-Rao bound. The Root Mean Square (RMS) Monte Carlo error for each fixed time instant is lower bounded by the Cramér-Rao bound,

$$\sqrt{\frac{1}{M} \sum_{i=1}^{M} \|x_t - \hat{x}_t^i\|^2} \gtrsim \sqrt{\operatorname{tr} P_t}$$

where $\hat{x}_t^i$ is the one step ahead prediction of the states at time t in Monte Carlo run i.

Figure 3 shows a 300 samples long track over a part of the terrain map from Figure 2. The aircraft travels from right to left. The Cramér-Rao bound and the obtained RMS error from 1000 Monte Carlo simulations with the PMF is shown in Figure 4. The PMF is initiated with a prior with large support using a low resolution of the grid, this yields far from optimal performance during the first half of the simulations. However, as the grid resolution increases the RMS error decreases and when the PMF resolution reaches a steady state level, the filter performance reaches the optimal bound. Note also that both the bound and the performance depend on the terrain variations along the true track. The
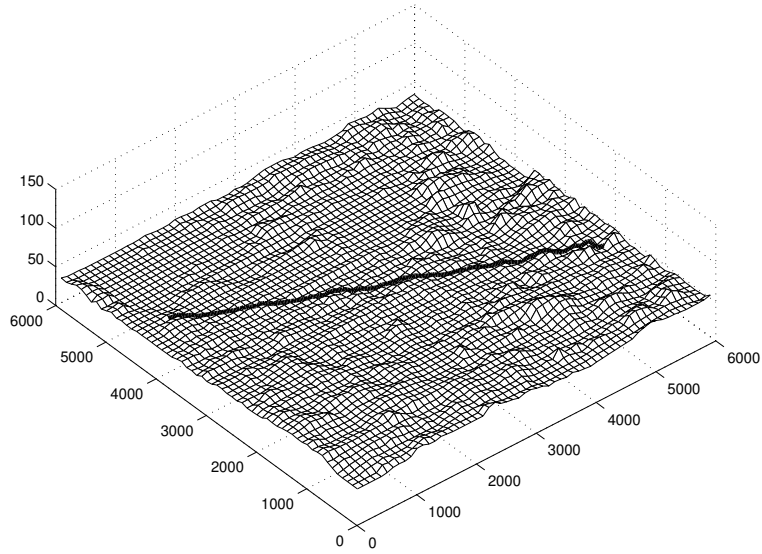
13

Figure 3: The simulation area and the true track, axes are labeled in meters.
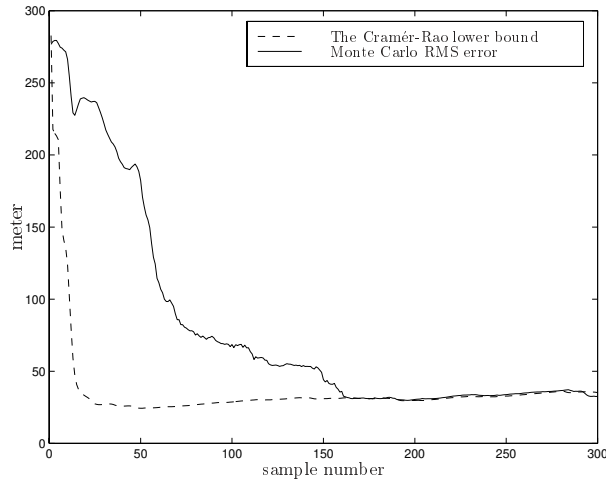

Figure 4: Monte Carlo root mean square error compared with the Cramér-Rao bound.

Monte Carlo evaluation above shows that the grid refinement method used in the PMF works very well and that the filter achieves the optimal performance

when the grid is chosen dense enough.

4 Simulation Evaluation

The terrain map used in these simulations is the same terrain database over a part of Sweden as is shown in Figure 2. A contour plot over this terrain map and the true simulated aircraft flight path is shown in Figure 5. The aircraft starts
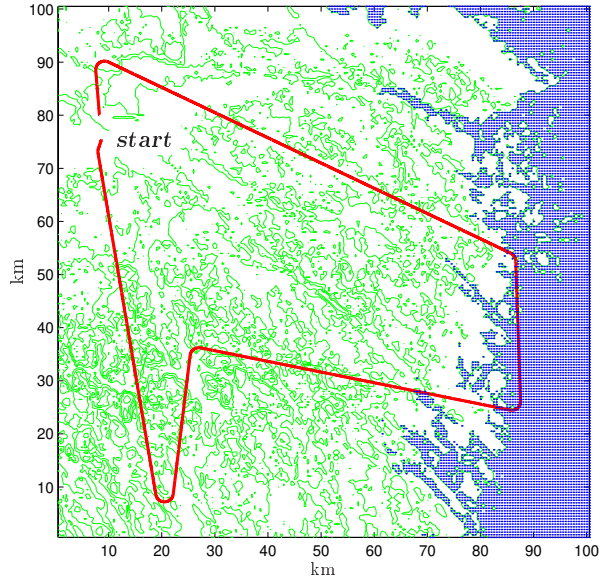

Figure 5: The simulation track over the terrain database.

heading south, after a few turns over the rough part of the map it flies over a part of the Baltic sea and then turns back and completes the counter-clockwise lap. The simulated aircraft track, the INS measurements and the radar altimeter measurements have all been generated in an advanced realistic simulator used by the navigation systems development department at Saab Dynamics. The track has a duration of 25 minutes and is sampled at a rate of 10 Hz. The aircraft has an average speed of Mach 0.55, and the manoeuvres are simulated as coordinated turns.

The INS position estimate $\hat{x}_0$ is initiated with an error of 1000 m in both

the north and the east direction. The prior density is chosen as a Gaussian distribution centered at the erroneous INS estimate

$$p(x_0) = \mathrm{N}(\hat{x}_0, 1000^2 I_2). \tag{11}$$

The initial grid resolution used in the PMF to sample this function is $\delta = 200$ m. The dead-reckoning drift in the INS is simulated as a constant bias of 1 m/s in each channel. The distribution used in the algorithm to model this drift is Gaussian $p_{v_t}(\cdot) = \mathrm{N}(0, 4I_2)$. The choice of Gaussian distributions has proven successful in the simulations but any other suitable distribution that better models the position drift and the initial uncertainty may be used. There is no restriction in the PMF to Gaussian noises: the only assumption is that the noise can be regarded as white.

Different sensor models are used when generating the simulated measurements. Depending on the terrain category beneath the aircraft at the measuring instant, both the bias and the variance of the radar altimeter are adjusted. For example, flying over dense forest the radar altimeter has a bias of 19 m with a large variance. Additional noise is added to the measurement to simulate that the radar altimeter measurement performance degrades with increasing ground clearance distance. The density used to capture these effects in the PMF algorithm is a mixture of two Gaussian distributions,

$$p_{e_t}(\cdot) = 0.8\,\mathrm{N}(0, 2) + 0.2\,\mathrm{N}(15, 9).$$

This choice can be interpreted as on the average every fifth measurement being biased due to reflection in trees or buildings. The truncation and resampling parameters used in the PMF are,

$$\varepsilon = 10^{-3}, \qquad N_0 = 1000, \qquad N_1 = 5000.$$

The simulation result from the first three recursions is depicted in Figure 6. Starting with the Gaussian prior (11), the first measurement amplifies the probability in several regions and removes samples of low probability. After the second recursion, the grid resolution is increased to 100 m and the third recursion removes even more samples and a single peak of the density shows the most probable aircraft position while the uncertainty still is rather large. The bounding box indicating the support of the prior is shown as a comparison with the
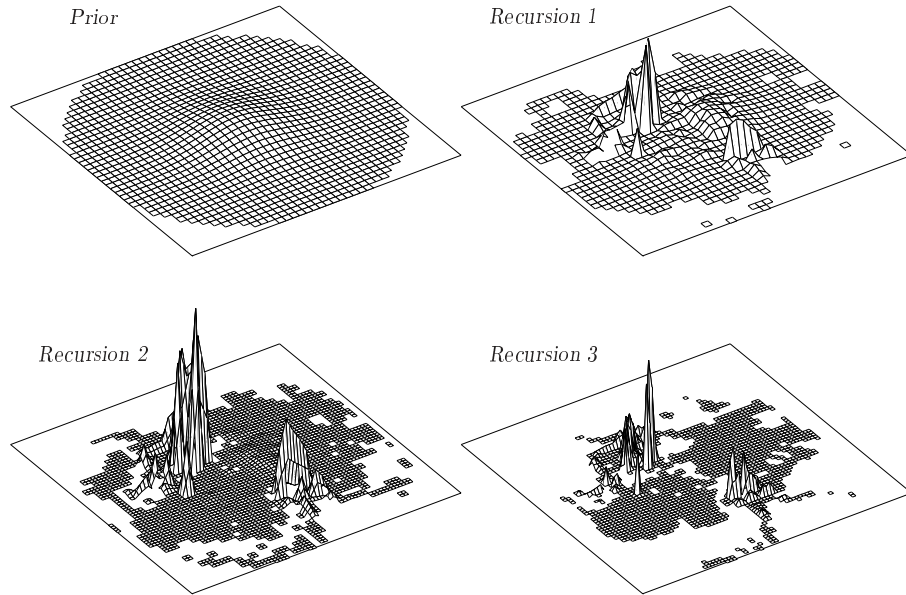
16

Figure 6: The first three recursions of the algorithm

support of each of the filter densities. The irregular shapes of these densities shows how the unstructured nonlinear terrain gives a filter density which is hard to approximate with smooth functions or local linearizations. Figure 7 shows the estimation error along the simulation track. Here it is obvious that the performance depends on the covered terrain. The error converges rapidly from the initial error of more than 1 km down to an error less than 30 m. When the aircraft reaches the Baltic sea the measurements have little information and the error increases with the drift of the INS. Once back over land, the estimate accuracy increases and a trend towards worse performance is visible when the aircraft covers the low informative areas of the map during the final part of the lap. The resolution of the grid is automatically adjusted and varies between 200 m and 0.78 m along the simulation track. A common navigation performance parameter is the circular error probable (CEP) which is the median of the position error. The simulation yields a median error of 12.2 m CEP. As a
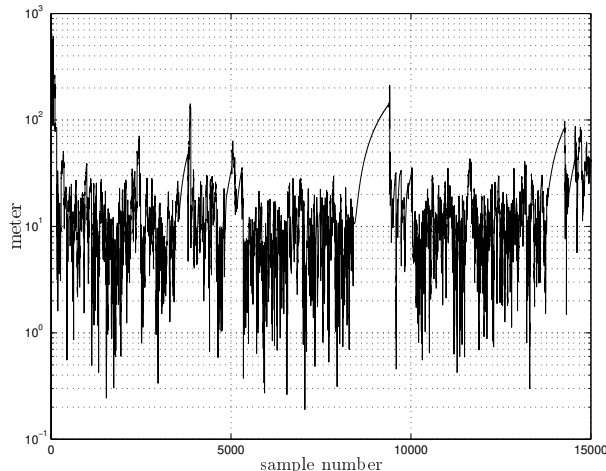
17

Figure 7: Estimation error along the simulation track, in logarithmic scale.

comparison, in [12] an error of 50 m CEP is reported and in [6] a value of 75 m is obtained. It should be remarked that both these values are found during field tests and not simulations.

5 Conclusion

The performance of terrain navigation depends on the size of the terrain gradient in the area. The point-mass filter described in this work yields an approximate Bayesian solution that is well suited for the unstructured nonlinear estimation problem in terrain navigation. It recursively propagates a density function of the aircraft position. The shape of the point-mass density reflects the estimate quality, this information is crucial in navigation applications where estimates from different sources often are fused in a central filter. The Monte Carlo simulations show that the approximation can reach the optimal performance and the realistic simulations in Section 4 show that the navigation performance is very high compared with other algorithms and that the point-mass filter solves the recursive estimation problem for all the types of terrain covered in the test.

The main advantages of the PMF is that it works for many kinds of nonlinearities and many kinds of noise and prior distributions. The mesh support and

18

resolution are automatically adjusted and controlled using a few intuitive design parameters. The main disadvantage is that it cannot solve estimation problems of very high dimension since the computational complexity of the algorithm increases drastically with the dimension of the state space. The implementation used in this work shows real-time performance for two dimensional and in some cases three dimensional models, but higher state dimensions are usually intractable.

6   Acknowledgment

References

[1] B.D.O. Anderson and J.B. Moore. *Optimal Filtering*. Prentice Hall, Englewood Cliffs, NJ, 1979.

[2] W. Baker and R. Clem. Terrain contour matching [TERCOM] primer. Technical Report ASP-TR-77-61, Aeronautical Systems Division, Wright-Patterson AFB, Aug. 1977.

[3] N. Bergman. *Bayesian Inference in Terrain Navigation*. Linköping Studies in Science and Technology. Thesis No 649, 1997.

[4] N. Bergman. On the Cramer-Rao bound for terrain-aided navigation. Technical Report LiTH-ISY-R-1970, Dept. of EE, Linköpings University, 1997.

[5] N. Bergman, L. Ljung, and F. Gustafsson. Point-mass filter and Cramer-Rao bound for terrain-aided navigation. In *Proc. 36:th IEEE Conf. on decision and control*, pages 565–570, 1997.

[6] D. Boozer and J. Fellerhoff. Terrain-Aided Navigation Test Results in the AFTI/F-16 Aircraft. *Journal of The Institute of Navigation*, 35(2):161–175, 1988.

[7] R.S. Bucy and K.D. Senne. Digital synthesis of nonlinear filters. *Automatica*, 7:287–298, 1971.

[8] R. Enns and D. Morrell. Terrain-aided navigation using the Viterbi algorithm. *Journal of Guidance, Control and Dynamics*, 18(6):1444–1449, 1995.

[9] J.P. Golden. Terrain contour matching (TERCOM): a cruise missile guidance aid. In T. F. Wiener, editor, *Image Processing for Missile Guidance*, volume 238, pages 10–18. SPIE, 1980.

[10] A.J. Henley. Terrain aided naviagtion – current status, techniques for flat terrain and reference data requirements. In *Proc. of IEEE Pos., Loc. and Nav. Symp. (PLANS)*, Las Vegas, 1990.

[11] C. Hewitt. The use of terrain databases for avionic systems. In *IEE Colloquium on Terrain databases and their use in navigation and collision avoidance*, 1995.

[12] J.A. Hollowell. Heli/SITAN: A terrain referenced navigation algorithm for helicopters. Las Vegas, 1990. IEEE Pos., Loc. and Nav. Symp. (PLANS).

[13] L.D. Hostetler. Optimal terrain-aided navigation systems. In *AIAA Guidance and Control Conference*, Palo Alto, CA, 1978.

[14] M. Kayton and W. Fried, editors. *Avionics Navigation Systems*. 2nd edition, 1997.

[15] S.C. Kramer and H.W. Sorenson. Bayesian parameter estimation. *IEEE Transactions on Automatic Control*, 33:217–222, 1988.

[16] G.M. Siouris. *Aerospace Avionics Systems*. Academic Press, 1993.

[17] H.W. Sorenson. Recursive estimation for nonlinear dynamic systems. In J. C. Spall, editor, *Bayesian Analysis of Time Series and Dynamic Models*, pages 127–165. Marcel Dekker inc., 1988.