Simulation of random events for air traffic applications

Stéphane Puechmorel, G Dufour, Romain Fèvre

Simulation of random events for air traffic applications[☆]

S. Puechmorel[a,*], G. Dufour[b], R. Fèvre[c]

[a] *ENAC - Unversité Fédérale de Toulouse*
7, Avenue Edouard Belin
FR-31055 TOULOUSE CEDEX
[b] *ONERA*
2 avenue Edouard Belin FR-31055 TOULOUSE CEDEX 4
[c] *Altys*
7 avenue Parmentier
Central Parc 2 - Bâtiment A
FR-31200 TOULOUSE

Abstract

Resilience to uncertainties must be ensured in air traffic management. Unexpected events can either be disruptive, like thunderstorms or the famous volcano ash cloud resulting from the Eyjafjallajökull eruption in Iceland, or simply due to imprecise measurements or incomplete knowledge of the environment. While human operators are able to cope with such situations, it is generally not the case for automated decision support tools. Important examples originate from the numerous attempts made to design algorithms able to solve conflicts between aircraft occurring during flights. The STARGATE project was initiated in order to study the feasibility of inherently robust automated planning algorithms that will not fail when submitted to random perturbations. A mandatory first step is the ability to simulate the usual stochastic phenomenons impairing the system: delays due to airport platforms or ATC and uncertainties on the wind velocity. The work presented here will detail algorithms suitable for the simulation task.

Keywords: Fast time traffic simulator, Gaussian field simulation, Air traffic management

1. Introduction

One of the most important source of uncertainties in the ATM system comes from the wind. When flying a given path, an aircraft can experience two kind of errors: a cross-track error that describes the lateral deviation from the intended trajectory and a longitudinal error that describes the distance between the current and the expected position at a given time. The figure 1 summarizes these two measures. Flight management systems (FMS) are able to deal efficiently with the cross-track error that can be reduced to very low values that are

[☆]This work was supported by the French National Research Agency (ANR) within the

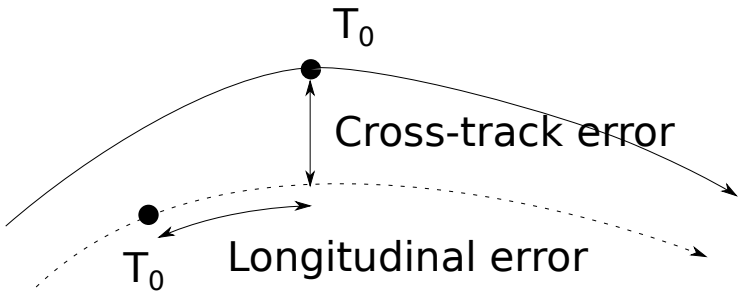

Fig. 1. Longitudinal and cross-track errors

generally not a concern, at least during en-route flight phase. Managing the longitudinal error relies on changing the aircraft velocity mainly by engine thrust adjustment. Such an operation impairs fuel consumption and may not even be possible as during cruise phase, aircraft are operated close to the maximal altitude, reducing the safe speed interval.

As a consequence, keeping an aircraft on a 4D trajectory, that is bound to a given position at a given time, requires a perfect knowledge of atmospheric conditions. Even in the context of future ATM systems like SESAR or Nextgen, the holy graal of full 4D trajectory does appear as feasible: requested time-to-arrival (RTA) at given predefined positions, known as 3D+t procedures, is a more realistic option. Nevertheless, due to the traffic increase, moving from airspace based operations to trajectory based ones appears mandatory. In this context, the control task will no longer be performed uniquely by human operators and automated tools will help in maintaining aircraft separated. To gain a wide social acceptance of the concept, systems must be proven safe, even under uncertainties on aircraft positions.

Several automated trajectory planners exist, [1] most of them being adapted from their counterparts in robotics, but are generally very sensitive to random perturbations. Navigation functions are of current use for autonomous robots and have been extended to air traffic problems [2]. They offer built-in collision avoidance and are fast to compute. Furthermore, relying basically on a gradient path following algorithm, it is quite easy to derive convergence properties. An interesting approach to design navigation functions is the use of harmonic potentials and recently bi-harmonic ones. The computation is reduced to an elliptic partial differential equation solving, that can be done readily using off-the-shelf software. The STARGATE project, funded by the french agency ANR, was initiated to extend harmonic and bi-harmonic navigation functions to a stochastic setting. Within this frame, a mean of simulating random events is required, As mentioned above, only the longitudinal error along track is a concern, and it

project STARGATE
*Corresponding author

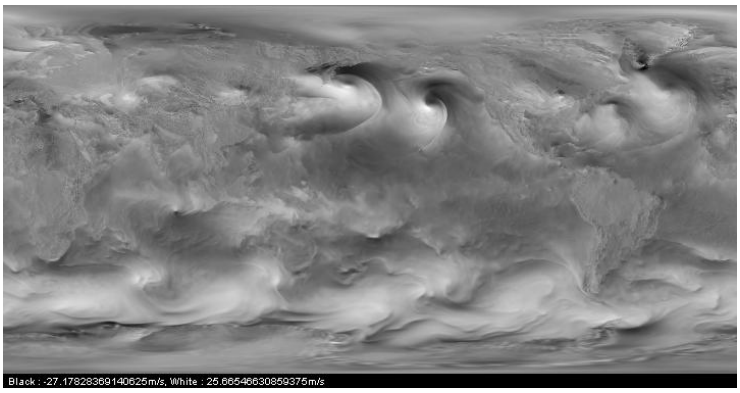

Fig. 2. A typical wind field (world coverage)

comes from two main sources:

- The airport departure delay and the ATC actions;
- The effect of the wind experienced along the flight path.

Airport or ATC delays may be somewhat anticipated, but an important variability still exists. Concerning the wind, meteorological models are not accurate enough to ensure a good predictability: the best possible description of a wind field is the sum of a mean component, generally taken to be the output of the atmosphere models, and a random perturbation.

The first part of this article will be devoted to the means of generating random wind fields, that will be processed within a fast time air traffic simulator. The resulting synthetic flight paths can be used to assess the performance of the conflict solving algorithms and to test the resilience of the planning schemes produced.

In a second part, a statistical analysis of airport and ATC delays will be presented, and a simple delay generator will be detailed.

Finally, some numerical implementation issues will be discussed.

2. Random wind field simulation

Wind fields exhibit generally some structure along with more chaotic behavior in some areas as shown on figure 2. Meteorological models forecast wind speed on a sampling grid, with angular increments that can be as low as 0.01° (e.g. the AROME model from the french weather service). The temporal resolution is not so high, typically in the hour range. Due to the requirements of future ATM systems, providers are working on an order of magnitude improvement, but such data is not yet available.

Even with the best available resolution, mesh size is in the order of 1Km, not enough to ensure a very high quality trajectory prediction: an interpolation error

will add to the predicted position. The same is true for the time interpolation point of view and both effects may be modeled as a random variable. Finally, inherent model inaccuracies can be incorporated, so that the wind field X may be represented as the sum of a deterministic mean value X_0 and a stochastic field η. While η is a vector valued field, it will be described in coordinates and only scalar fields will be considered in the sequel.

2.1. Spectral representation

It is classically assumed that η is an homogeneous Gaussian random field defined on $\mathbb{R}^d$, width $d = 2$ for planar fields and $d = 3$ for the general case. It is further assumed that η does not depend on time, or that the time evolution is slow enough compared to the time scale of the trajectory predictor.

It is characterized by its mean and covariance functions, respectively denoted μ, C and C satisfies the (spatial) stationarity condition:

$$\forall (x, y) \in \Omega,\, C(x, y) = C(x - y) \tag{1}$$

One of the most important property of Gaussian fields is related to finite samples of it.

Proposition 1. *For any finite sequence $(x_1, \ldots, x_n)$ of points in $\mathbb{R}^d$, the vector $(X(x_1), \ldots X(x_n))$ is Gaussian, with mean*

$$(\mu(x_1), \ldots, \mu(x_n))$$

and covariance

$$\Sigma = (C(x_i - x_j))_{i,j=1\ldots n}$$

The covariance function is positive definite:

$$\forall (a_1, \ldots a_n) \in \mathbb{R}^n,\, \forall (x_1, \ldots x_n) \in \Omega^n,\, \sum_{i,j} a_i a_j C(x_i - x_j) \geq 0 \tag{2}$$

The Bochner theorem gives a characterization of positive definite functions:

Theorem 1. *A continuous mapping $f \colon \mathbb{R}^d \to \mathbb{R}$ is positive definite if and only if it exists a finite positive measure F such that:*

$$\forall x \in \mathbb{R},\, f(x) = \int_{\mathbb{R}} e^{i2\pi\langle t, x\rangle} dF(t)$$

It is most of the time easier to check that a given mapping is positive definite through the use of 1 than with a direct proof. In the case of covariance functions, F is known as the spectral measure of the random field and when it admits a density θ, one can compute a numerical approximation of it using a discrete Fourier transform. This property is the key ingredient of many efficient random field generators. The spectral density θ can be shown to be even and positive.

2.2. Gaussian field generation

Algorithms for scalar Gaussian field generation can be direct or spectral. In the first case, the proposition (1) is invoked to generate Gaussian vectors on a mesh of points $(x_1, \ldots, x_N)$ that is generally structured as a rectangular, evenly spaced grid.

Proposition 2. *Let $u = (u_1, \ldots, u_N)$ be a sample of independent, normally distributed real numbers. Let v be a vector in $\mathbb{R}^N$ and C a symmetric, positive definite matrix. The vector $C^{1/2}u + v$ is Gaussian, with mean v and covariance C.*

Proof. It is a very classical result that comes at once from the properties of mathematical expectation. Since u has zero mean:

$$E[C^{1/2}u + v] = C^{1/2}E[u] + v = v$$

The covariance is then given by:

$$E[C^{1/2}ut^tC^{1/2}] = C^{1/2}E[uu^t]C^{1/2} = C^{1/2}C^{1/2} = C$$

The computation of the square root $C^{1/2}$ is not required: the Cholesky decomposition $C = L^tL$ will give the same result and is much easier to obtain. The algorithmic complexity of the process is dominated by the Cholesky decomposition, that is an $O(N^3)$ process. Once done, sampling the vector u and performing the combination $Lu+v$ is of order $O(N^2)$. In many cases, the covariance will quickly drop for distant pairs of points: the matrix C is often nearly sparse, and adapted algorithms for the Cholesky decomposition exist. However, this stage remains the bottleneck of the Gaussian field generation procedure.

Spectral algorithms rely on a fast Fourier transform to let computations occur in the frequency domain. In [3], some methods for sampling Gaussian fields based on this approach are detailed.

The general principle underlying these methods is the use of Bochner theorem and Ito isometry formula. While almost never stated in references related to numerical Gaussian field simulations, it is a very important point that may be used as basis to derive algorithms. The approach taken in this work is to directly approximate the spectral representation of the field instead of using it only as a convenient tool for lowering the computational cost. A benefit is that optimal quadrature rules may be used in the fourier domain to improve the reconstruction of the simulated field.

Let us first start with some basic facts about spectral representation of processes as exposed in the reference work [4].

Definition 1. *Let $\mathcal{P} = (\Omega, \mathcal{T}, P)$ be a probability space and let $L^2 = L^2(\omega\mathcal{T}, P)$ be the space of finite variance complex random variables over it. An orthogonal stochastic measure on $\mathcal{P}$ is a mapping μ from $\mathcal{T}$ to L^2 such that:*

- $\mu(\emptyset) = 0$

- *For all couples (A, B) in $\mathcal{T}$ such that $A \cap B = \emptyset$:*

$$\mu(A \cup B) = \mu(A) + \mu(B)$$

  P-almost surely.

- *It exists a measure m on $\mathcal{T}$ such that for all couples (A, B) in $\mathcal{T}$:*

$$E[\mu(A)\overline{\mu(B)}] = m(A \cap B)$$

The measure m occurring in the definition 1 is called the structural measure of μ The definition can be extended straightforwardly to vector valued, finite variance random variables by requiring that the covariance matrix $E[\mu(A)\overline{\mu(B)}]$ is a σ-additive function on $\mathcal{T}$ with diagonal entries ordinary measures.

Given a simple function $f = \sum_{i=1}^{n} a_i 1_{A_i}$, one can define its μ-integral as:

$$\int f(x)d\mu(x) = \sum_{i=1}^{n} a_i \mu(A_i)$$

which is an L^2 random variable by definition of μ. Given the fact that random variables in L^2 are limits of such functions, one can define the μ-integral of an arbitrary L^2 random variable.

A very important isometry theorem holds for μ-integrals.

Theorem 2. *Let f, g be L^2 random variables. Then:*

$$E\left[\int f(x)d\mu(x)\overline{\int g(x)d\mu(x)}\right] = \int f(x)\overline{g}(x)dm(x)$$

An important special case arises when μ is obtained from a left mean-square continuous process F with orthogonal increments. In such a case, one can construct μ on intervals $[a, b[$ by $\mu([a, b[) = F(b) - F(a)$ and extend to the Borel σ algebra on $\mathbb{R}$. The resulting μ-integral is a stochastic Stieljes integral. Extension to $\mathbb{R}^d$ can be done component-wise, using independent processes in each coordinate and take as the stochastic measure of rectangle the product of the stochastic measures of its sides. Itô integral is recovered when F is the Wiener process, the theorem 2 being the Itô isometry.

Theorem 3. *Let X be an complex stochastic process on $(\Omega, \mathcal{T}, P)$ with a well defined covariance function C_X. If it exists a measure space $(A, \mathcal{A}, m)$ and measurable functions $g\colon \Omega \times A \to \mathbb{C}$ such that:*

$$\forall(\omega_1, \omega_2) \in \Omega^2,\ C_X(\omega_1, \omega_2) = \int g(\omega_1, x)\overline{g(\omega_2, x)}dm(x)$$

Then it exists a stochastic orthogonal measure μ on $\mathcal{T}$ with structure measure m such that almost surely:

$$\forall\omega \in \Omega, X(\omega) = \int g(\omega, x)d\mu(x)$$

The theorems extends to vector-value stochastic processes (see [4]).

The following special case will be the one used in the sequel:

Theorem 4. *Let X be a complex vector valued continuous, stationary random process defined on $\mathbb{R}$ with zero mean. Then it exists a vector-valued stochastic orthogonal measure μ such that almost surely:*

$$X(t) = \int e^{it\xi} d\mu(\xi)$$

2.3. Principles of gaussian fields simulation

If C is the covariance function of the field that must be simulated, then it can be written according to 1 as:

$$[\forall x, y \in R^d, C(x,y) = \int_{\mathbb{R}^d} \exp(i\langle x-y, \xi\rangle) dF(\xi)$$

and if μ is absolutely continuous with respect to the Lebesgue measure as:

$$[\forall x, y \in R^d, C(x,y) = \int_{\mathbb{R}^d} \exp(i2\pi\langle x-y, \xi\rangle)\theta(\xi) d\xi \tag{3}$$

with θ a density. Let Z be the centered random field given by:

$$Z_x = \int_{\mathbb{R}^d} \exp(i2\pi\langle x, \xi\rangle)\theta^{1/2}(\xi) dB_\xi \tag{4}$$

with B_ξ the d-dimensional complex Wiener process. By the Ito isometry it comes;

$$E[Z_x Z_y] = \int_{\mathbb{R}^d} \exp(i\langle x, \xi\rangle) \exp(i\langle y, \xi\rangle)\theta(\xi) d\xi \tag{5}$$

$$= \int_{\mathbb{R}^d} \exp(i2\pi\langle x-y, \xi\rangle)\theta(\xi) d\xi = C(x,y) \tag{6}$$

Finally, the process $Z_x + \mu(x)$ has covariance C and mean μ as required.

For numerical simulations, it is needed to generate samples of the above stochastic processes at fixed positions, that may be done using a very simple approximation of the integral as a finite sum. First of all, the domain of integration has to be reduced to a bounded region Ω of $\mathbb{R}^d$ in order to make it amenable to numerical implementation. Since in the spectral representation the density θ is even, it makes sense to let Ω be a product of centered rectangles: $\Omega = \prod_{i=1}^d [-a_i, a_i]$ (here $d = 2$ or $d = 3$). An elementary cell in the grid, $\mathcal{C}_k$, indexed by the vector $k = (k_1, \ldots, k_d)$ with integer coordinates, will be written accordingly as:

$$\mathcal{C}_k = \prod_{i=1}^d [\xi_k^i, \xi_{k+1}^i[$$

where for each i, $-a_i = \xi_0^i < \xi_1^i < \cdots < \xi_{N_i}^i = a_i$ is a subdivision of the interval $[-a_i, a_i]$. Please note that the index coordinates $k_1, \ldots, k_d$ are for each i in the set $0, \ldots, N_i - 1$. The discretization of the integral in (4) over the partition given by the cells $\mathcal{C}_k$ using the rectangle quadrature formula gives:

$$Z_x = \sum_{k_1=0,\ldots,k_d=0}^{N_0-1,\ldots,N_d-1} \exp\left(i2\pi\langle x, \xi_{k_1,\ldots,k_d}\rangle\right) \theta^{1/2}(\xi_{k_1,\ldots,k_d}) V_{k_1,\ldots,k_d} \tag{7}$$

with:

$$V_{k_1,\ldots,k_d} = \prod_{i=1}^{d} \left(W_i(\xi_{k_{i+1}}^i) - W_i(\xi_{k_i}^i) \right) \tag{8}$$

In the computation of the volume element (8), $W_1, \ldots, W_d$ are independent one-dimensional Wiener processes. Due to the fact that W_i has normally distributed increments, it appears that $V_{k_1,\ldots,k_d}$ follows a d-dimensional centered Gaussian distribution with diagonal covariance matrix and variance $\xi_{k_{i+1}}^i - \xi_{k_i}^i$ in dimension i: generating samples according to such a law is easily done using for example the Box-Muller algorithm [5].

In practice, only regular grids are used, and we may further assume that discretization steps are equal to a fixed value δ in each direction. The random number generator is then calibrated once for all to draw samples according to a $\mathcal{N}(0, \delta)$. Please note that the algorithm takes place in the Fourier domain, so characteristic dimensions must be inverted when going to the space domain, as it will be made more precise in the sequel, when dealing with numerical implementation.

2.4. Finding a covariance function

Model covariance functions must comply with the requirements already mentioned above, namely be a symmetric and of positive type function. A simple approach is to fit a standard function belonging to this class on measured data. Classical choice are exponential functions, either in spatial or the Fourier domain: in such a case the transformed function in the dual domain will be a slowly decreasing mapping, behaving roughly as $\| \bullet \|^{-2}$ at infinity. Another option is to use a gaussian function, that is of positive type and exhibits fast decrease in both the spatial and Fourier domain. Simulated fields with gaussian covariance functions are given in figures.

A more satisfying approach will be to start with real observations then estimate directly the covariance from them. However, the resolution of the available meteorological models, especially at the cruise altitude of commercial airliners, is a major issue, so that its implementation is postponed to a future work.

3. Airport and ATC delays: a statistical study

For the purpose of the STARGATE project, only a coarse modeling is needed for the airport delays, as simulations involving ground side are done at a strategical time horizon, namely weeks to days from real departure dates. Given the

uncertainties on the weather at this time scale, only very conservative plannings may be designed, that will serve as a basis for reference business trajectories negotiation. A statistical study was conducted on sample data coming from flight plans data in France over the years 2012-2015. A python parser was designed in order to transform the original data in COURAGE format (used by the french civil aviation authorities) into a csv file readable in R. The code is made available at the project website [6]. A simple pre-processing step is needed to get rid of aberrant values due to flights erroneously classified on a day, while departing one day before or after. This situation occurs on long-haul flights when part of the trajectory is made over the french airspace, but departure or arrival take place on another day. All flights with high delays are thus processed in order to check for such an event and are discarded from the initial set of samples. Furthermore, negative delays may occur but are uncommon and not useful for the application in mind: only flights with positive ones have been retained in the final sample. The density histogram on the cleaned data is shown on figure (3) From classical queuing theory, it is common to fit an exponential distribu-

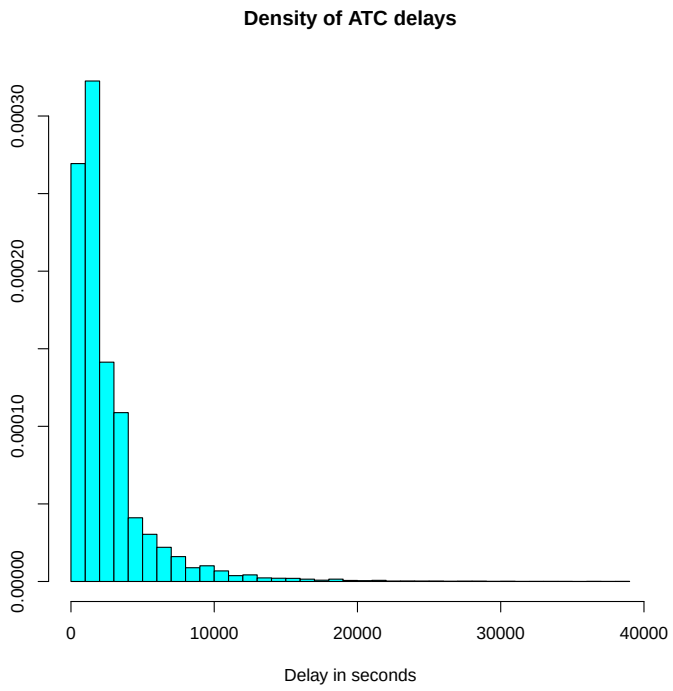

Fig. 3. Density of delays

tion on delays. Based on the histogram, other candidates are the Weibull and gamma distributions. Using the function `fitdistr` from the R package MASS, the maximum likelihood estimates in table 1 were found. The resulting densities

Distribution	parameters
Exponential	rate = 3.85e-4
Gamma	shape = 1.47, rate = 5.66e-4
Weibull	shape = 1.13, scale = 2730

Table 1: Maximum likelihood parameters estimates

are plotted against the sample histogram on figure (4) Only the initial part of

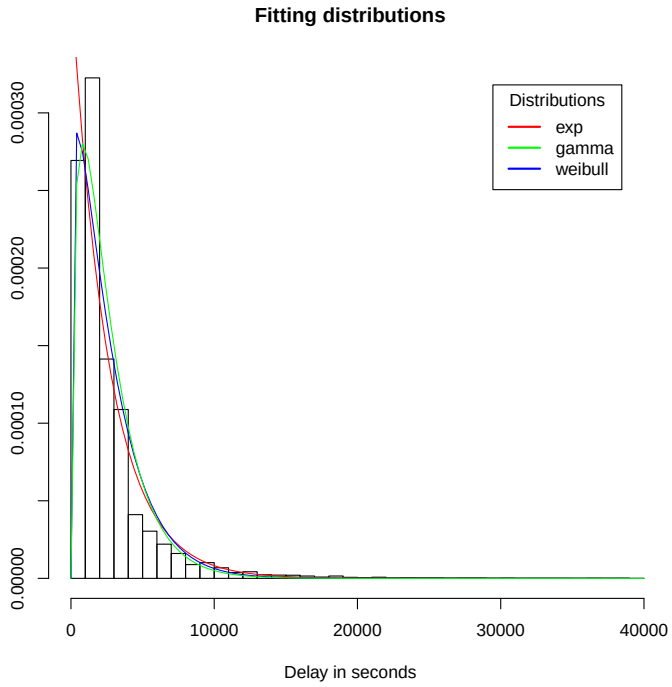

Fig. 4. Maximum likelihood fits

the curve makes a real difference, especially for the exponential distribution that will not vanish at the origin. For the simulation purposes and since plausible delays are to be generated, the exponential distribution was retained, although not the best fitted to observations. The reasons for such a choice are:

- The exponential distribution yields easier theoretical derivations. It is not a real issue for simulations, but the stochastic delay model is intended to be used also for resilient automatic conflict resolution algorithms. In this context, the ability to derive provable results is mandatory.
- The initial part of the empirical histogram corresponds to small delays which are often inaccurately reported. It makes no real difference from an

operational point of view to assume that very small delays exist.

The well known absence of memory of the exponential law is questionable in the context of airport or ATC delays as one can expect them to be highly correlated in time. This is true at small time scales, say hours, but no longer verified at the larger time horizons considered in the present work.

The generation of exponentially distributed values is easy using either a transformation or the fast ziggurat algorithm.

4. Numerical implementation

The purpose of this section is to detail the way random wind field are generated. It is split into two parts: acquisition of weather data, with an emphasis on publicly accessible sources and stochastic simulation.

4.1. Weather data

The overall simulation process starts with an prior information about the deterministic component of the field. It is available from weather services as a grid of values computed from a numerical model. Depending on the reference time, real measurements may be part of the data, but will not cover the entire grid: an interpolation is realized implicitly by the model used. The french weather agency provides two ways for accessing the data:

- Direct download of files encoded in the GRIB format [7].
- Access through a web service complying with the European INSPIRE directive [8].

The highest resolutions can be obtained only using the first procedure. The table (2) summarizes the data downloadable with the associated grid spacing and coverage, expressed in latitude-longitude coordinates. Nearly all atmospheric parameters are available. The complete description of the AROME and ARPEGE models is available at [9]. The figure (5) is an example of wind data acquired from the 0.01° AROME model.

Area	grid step	model	coverage (lat-lon)
Europe	0.1°	ARPEGE	72N 20N 32W 42E
Globe	0.5°	ARPEGE	World
France	0.01°	AROME	55.4N 37.5N 12W 16E
France	0.025°	AROME	53N 38N 8W 12E

Table 2: Weather data downloadable from Meteo France.

Unfortunately, high resolution data cannot be accessed from the web service at the time where this article was written, only the 2.5° world coverage was implemented. It is expected that all models will be soon ported to this platform,

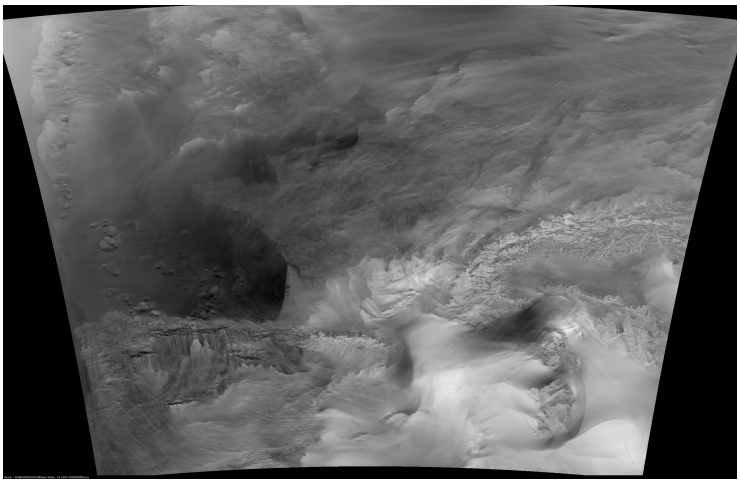

Fig. 5. Wind v-component. 0.01° resolution

which will then become the preferred data source. Due to the resolution limitations and the fact that fine step samples are needed to accurately implement the random field generation, only the outputs from the AROME model were used in the present work. The relevant atmospheric parameters were the u, v wind components, associated respectively to projections onto local longitude and latitude coordinates and with an altitude-pressure of 250 hPa, roughly equal to the flight level 340, which is the mean cruise altitude for commercial aircraft (one must note that resolution is reduced at this altitude compared to the original 0.01°).

4.2. Gaussian field simulation

As indicated in section 2, simulation will be conducted by the mean of a spectral representation. The fast Fourier transform (FFT) is an $O(n \log n)$ algorithm for efficiently computing sums like the one appearing in (7). The software library used for that purpose is FFTW [10, 11], a very efficient implementation that is able to operate on sequences of arbitrary lengths[1].

Let us first recall the definition of a discrete Fourier transform (DFT), that is computed by the FFT algorithm. Given a sequence $x_0, \ldots, x_{N-1}$, its DFT is the sequence $\hat{x_0}, \ldots, \hat{x_{N-1}}$ given by:

$$\hat{x_k} = \sum_{j=0}^{N-1} x_j e^{-i2\pi \frac{jk}{N}} \tag{9}$$

Looking at the expression 9, it appears immediately that $\hat{x_k}$ is a periodic sequence, of period N. Furthermore, if the DFT is interpreted as an approxi-

[1] The original FFT algorithms requires lengths in powers of two

mation of a Fourier integral, then the domain of integration has length 1, due to the j/N appearing in the complex exponential. The inverse DFT is defined pretty much the same way:

$$x_k = \frac{1}{N} \sum_{j=0}^{N-1} \hat{x}_j e^{i2\pi \frac{jk}{N}} \tag{10}$$

The DFT can be used to approximate an integral if considered as a Riemann sum. In such a case, it is easy to see that if the points $x_j, j = 0 \dots N-1$ are considered as samples at positions jT/N in the interval $[O, T[$, so that the time period is T, then the corresponding sample positions in the frequency domain are located at $k/T, k = 0 \dots N-1$. Introducing the so-called sampling frequency $f_s = N/T$, the samples in the frequency domain are expressed as $kf_s/N, k = 0 \dots N-1$. Going back to formula (8), the points ξ_j^i appearing in the expression correspond to a subdivision of the interval $[-f_s^i/2, f_s^i/2]$ where f_s^i is the sampling frequency in the dimension i (with corresponding number of samples N_i). This fact is important and often overlooked: the frequency domain depends on the sampling rate, and thus if the time (or spatial) interval is fixed, on the number of discretization points. On the other hand, the frequency resolution, that is the difference between two samples in the Fourier domain depends only on T, the length of the time interval. As a consequence, the respective variances of the Gaussian increments occurring in the expression (8) are $1/T_i$, where T_i is the length of the spatial domain in dimension i, and are not dependent on the number of sampling points. The same remark applies for the function θ appearing in (8) where care must be taken to use the points $\xi_i = (k - N_i/2)/T_i, k = 0 \dots N_i$. When using the inverse DFT for computing, it is important to note that the term $1/N$ present in the expression must be scaled out in order to get an approximate integral. Finally, the procedure described above will generated a complex valued random field, which is not needed in practice, unless one want to get in one step two independent samples. It is possible to get a real valued field by generating only half of the Gaussian random variables and taking the complex conjugate of them for the remaining ones. Gathering things, a 2D-field generation algorithm on a evenly spaced grid with N_x, N_y points in each coordinate, can be summarized as indicated below:
The algorithm 1 returns a field with energy proportional to the product $f_x f_y$. This can be understood intuitively by recalling that the DFT approximates an integral over $[-f_x/2, f_x/2] \times [-f_y/2, f_y/2]$. If a unit energy is needed, then an inverse scaling must be applied or a normalization performed: this last options is preferred as it removes any inaccuracy coming from the finite precision computation of the FFT, at the expense of a higher computational cost.

5. Conclusion and future work

The present work describes means of simulating random events occurring in the context of the air traffic system. ATC and airports delays can be modeled

Algorithm 1 2D Gaussian field generation

Require: M is a $N_x \times N_y$ complex matrix
Require: f_x, f_y are the respective sample frequencies in x, y
Require: θ is the required spectral density
Require: $N(\mu, \sigma)$ generates independent random normally distributed real numbers with mean μ and variance σ
  for $i = 0 \dots N_x - 1$ **do**
    for $j = 0 \dots N_y - 1$ **do**
      if $i == 0$ or $j == 0$ or $i == N_x/2$ or $j == N_y/2$ **then** ▷ Endpoints are handled differently: they have a vanishing imaginary part
        $\Re M(i,j) \leftarrow N(0, f_x * f_y)$
        $\Im M(i,j) \leftarrow 0$
      else
        $\Re M(i,j) \leftarrow N(0, f_x * f_y) * \theta^{1/2}(\xi_i, \xi_j)$
        $\Im M(i,j) \leftarrow N(0, f_x * f_y) * \theta^{1/2}(\xi_i, \xi_j)$
        $M(N_x - i, N_y - j) = \overline{M(i,j)}$
      end if
    end for
    $X \leftarrow N_x * N_y * IFFT(M)$ ▷ X is the generated real field
  end for

on an exponential distribution, taking into account the large time horizon considered. For the meteorological uncertainties, a simple random Gaussian field model is assumed. Generating such samples can be done efficiently by the mean of fast Fourier transforms. An approach based on the spectral representation of the processes and discretization of it in the Fourier domain is introduced, giving final algorithms similar to the ones described in the literature, but relying on a different principle. Due to the flexibility of the spectral representation, a future work will investigate the use of alternative functions for the covariance expansion. For example, wavelets may be considered to allow a multiscale generation of fields, allowing different characteristic lengths to be considered.

[1] C. Goerzen, Z. Kong, B. Mettler, A survey of motion planning algorithms from the perspective of autonomous uav guidance, Journal of Intelligent and Robotic Systems 57 (1) (2009) 65–100. doi:10.1007/s10846-009-9383-1.
URL http://dx.doi.org/10.1007/s10846-009-9383-1

[2] G. P. Roussos, K. J. Kyriakopoulos, Completely decentralised navigation functions for agents with finite sensing regions with application in aircraft conflict resolution, in: CDC, 2011.

[3] A. Lang, J. Potthoff, Fast simulation of gaussian random fields, Monte Carlo Meth. and Appl. 17 (2011) 195–214.

[4] I. Gikhman, A. Skorokhod, Introduction to the Theory of Random Pro-

cesses, Dover books on mathematics, Dover Publications, 1969.
URL https://books.google.fr/books?id=yJyLzG7N7r8C

[5] G. E. P. Box, M. E. Muller, A note on the generation of random normal deviates, Ann. Math. Statist. 29 (2) (1958) 610–611. doi:10.1214/aoms/1177706645.
URL http://dx.doi.org/10.1214/aoms/1177706645

[6] Open source code repository for the stargate project.
URL https://stargate.recherche.enac.fr/

[7] Public data downloading from meteo france.
URL https://donneespubliques.meteofrance.fr/?fond=rubrique&id_rubrique=51

[8] Meteo france web service.
URL https://donneespubliques.meteofrance.fr/?fond=geoservices&id_dossier=14

[9] Arpege and arome models manuals.
URL https://donneespubliques.meteofrance.fr/client/document/description_parametres_modeles-arpege-arome-v2_185.pdf

[10] Fftw website.
URL http://www.fftw.org/

[11] M. Frigo, S. G. Johnson, The design and implementation of FFTW3, Proceedings of the IEEE 93 (2) (2005) 216–231, special issue on "Program Generation, Optimization, and Platform Adaptation".