Stochastic Particle Method Applied to Local Wind Simulation
Antoine Rousseau, Frédéric Bernardin, Mireille Bossy, Tamara Salameh, Philippe Drobinski

To cite this version:

HAL Id: inria-00172503
https://inria.hal.science/inria-00172503
Submitted on 17 Sep 2007

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L’archive ouverte pluridisciplinaire HAL, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d’enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.
Stochastic particle method applied to local wind simulation

Antoine Rousseau, Frédéric Bernardin, Mireille Bossy
FirstName.LastName@inria.fr
INRIA, OMEGA Project, Sophia Antipolis, France
Tamara Salameh, Philippe Drobinski
LMD, Ecole Polytechnique, Palaiseau, France.

Abstract

A Lagrangian stochastic model is introduced in order to describe the local behavior of the wind. Based on some MM5 data, and thanks to particles driven by stochastic differential equations, we propose a numerical method allowing to improve the MM5 simulations at small scales, without requiring too much additional computational cost.

1. Introduction

The aim of this work is to propose a numerical method to improve the wind forecasting at small scales. Up to now, solvers like MM5 (developed at NCAR and PSU, see [3] and [4]) have a computational limit in both horizontal and vertical ranges. Namely, for a computation on the whole Mediterranean basin, it is quite difficult, if not hopeless, to run MM5 with an horizontal resolution finer than 1km. Thanks to some numerical tools based on particle methods, we wish to improve significantly this resolution. We recall that the computations that we consider are local and concern a small part of the MM5 computational domain. For this reason and since particle methods do not require any stability (CFL) condition, the CPU time should remain sufficiently small (compared to what MM5 requires).

2. Numerical algorithm and related issues

The numerical tool that we consider is based on the usual MM5 computations. Namely, MM5 will guide our simulations, since it both provides initial data and boundary conditions. This is a one way coupling so far, but we think of plugging the local information back into MM5 in future works.

Consider a 3D box (for example a cell of MM5 mesh) where the three components \((u, v, w)\) of the velocity field are given on each side. We use a Lagrangian approach, and adopt the point of view of a generic fluid particle, to get some information at smaller scales (inside the box). To this aim, we run a Particle In Cell (PIC) numerical algorithm, which consists in splitting our domain into cells \(C_i\) whose dimensions are \(\Delta x, \Delta y\) and \(\Delta z\), in which we drop \(K\) particles having Langevin dynamics (see Fig. 1). Then, we compute some local characteristics of the fluid by averaging the corresponding particle attributes in each cell.

The Lagrangian stochastic model that we study is borrowed from S.B. Pope [6] and describes the behavior of our particles in the considered domain \(D \subset \mathbb{R}^3\). We consider a generic fluid particle, and denote by \((X_t, U_t, \omega_t)\) its position, velocity and turbulence frequency. The model consists in writing a stochastic differential equation (SDE) that describes the dynamic of the pro-
mean-reverting terms of the following form:

\[
dX_t = U_t \, dt, \\
dU_t = -\frac{1}{\rho} \nabla \langle \mathcal{P} \rangle (t, X_t) \, dt + D_U \, dt + B_U \, dW_t, \\
d\omega_t = D_\omega \, dt + B_\omega \, dW_t' \tag{2.1b}
\]

where \((W_t)\) and \((W_t')\) are respectively 3D and 1D independent Brownian motions.

The drift coefficients \(D_U\) and \(D_\omega\) depend on some mean-reverting terms of the following form:

\[
U_t - \mathbb{E}(U_t | X_t = x) \big|_{x=X_t} = U_t - \langle U \rangle(t, X_t) \\
\omega_t - \mathbb{E}(\omega_t | X_t = x) \big|_{x=X_t} = \omega_t - \langle \omega \rangle(t, X_t),
\]

where \(\mathbb{E}\) denotes the mathematical expectation under the underlying probability space \((\Omega, \mathcal{F}, \mathbb{P})\), and \(\mathbb{E}(U_t | X_t = x)\) is the conditional expectation of the velocity \(U_t\) at the position \(X_t = x\). In this model, the mean velocity field \(\mathbb{E}(U_t | X_t = x)\) is assimilated to the Reynolds averaged Eulerian velocity \(\langle U \rangle(t, x)\). Idem for the turbulent frequency \(\omega_t\). These conditional expectations are computed thanks to Monte-Carlo methods, and thus \(K\) must be chosen sufficiently large to insure the convergence of the MC approximations in each cell \(C_i\):

\[
\langle U \rangle_i = \sum_{k=1}^{K} \mathbb{I}_{X_k \in C_i} U_k \tag{2.2}
\]

The diffusion coefficients \(B_U\) and \(B_\omega\) depend on \(\langle \omega \rangle(t, X_t)\) and the two first stochastic moments of the velocity. In particular, the coefficient \(B_\omega\) is such that the stochastic process \(\omega_t\) is nonnegative (see [2] for more details).

We supplement the SDEs (2.1) with the following continuity equation:

\[
\frac{\partial \rho}{\partial t} + \text{div}(\rho U) = 0, \tag{2.3}
\]

which is insured thanks to the two following constraints:

\[
\rho(t) = \text{cst} = \rho_0, \tag{2.4a}
\]
\[
\text{div}(U) = 0. \tag{2.4b}
\]

Finally, we recall that the particle dynamics in (2.1) must be constrained at the boundary of the computational domain: the mean velocities at the boundary are imposed by the MM5 values (MM5 guideline):

\[
\forall t \geq 0, \forall x \in \partial D, \quad \langle U_i \rangle(t, x) = V_0(t, x) \tag{2.5}
\]

The system (2.1) describes the evolution of \((X_t, U_t, \omega_t)\), which corresponds to the particle position, its velocity, and its turbulence frequency. These quantities are of Lagrangian type since the information is attached to the particle. On the contrary, the Eulerian quantities, that appear in the equations above, mainly \(\langle U \rangle(t, x)\) and \(\langle \omega \rangle(t, x)\), correspond to averaged quantities. They are computed on each cell \(C_i\) by averaging the corresponding values of the particles that belong to \(C_i\):

\[
\langle U \rangle(t, x) = \langle U \rangle_i, \quad \forall x \in C_i.
\]

The main steps of our numerical algorithm are the following (from time \(t_n = n \Delta t\) to time \(t_{n+1} = t_n + \Delta t\)). Given the values of the processes \((X^n, U^n, \omega^n)\) at time \(t = t_n\), we:

1. compute new values of velocity and turbulence frequency. To this aim, we compute \((\bar{U}^{n+1}, \omega^{n+1})\) thanks to equation (2.1c) and

\[
dU_t = D_U \, dt + B_U \, dW_t. \tag{2.6}
\]
Compared to original equation (2.1b), the latter equation does not take the pressure gradient into account, but this is done in the correction step for the velocities (see item 4 below).

The simulation of such stochastic differential equations inside $D$ is rather classical and is done thanks to a Euler scheme (e.g. [7]). A particular attention must be taken when the particles reach the boundary, in order to take into account the constraint (2.5).

2. move particles, that is compute $\tilde{X}^{n+1}$ in order to verify equations (2.1a). Some particles can go out the domain according to outward MM5 velocity at the boundary. In that case, introduction of new particles at inward boundary must be performed.

3. modify the positions from $\tilde{X}^{n+1}$ to $X^{n+1}$, so that (2.4a) is satisfied. This step is not classical at all and concerns the particle positions. The aim is to correct the particle positions in the domain such that:
   - their density is uniform (i.e. the number of particles is the same in each cell).
   - the “transport cost” is minimum.

This is a problem of discrete optimal transportation, which is known to be nonlinear and numerically very difficult in dimension 3 (see [1], [8], [5]), whereas the one-dimensional corresponding problem is linear, and easy to handle from the numerical viewpoint.

We thus introduce a method based on the simple case of dimension 1, for which the optimal transportation simply consists in sorting the particles (see e.g. Brenier [1]). We split the domain $D$ in what we call particle-tubes (see Fig. 2 below), defined by:

$T^x_{j,k} = \{(x, y, z) \in D, (j-1)\Delta y \leq y \leq j\Delta y, (k-1)\Delta z \leq z \leq k\Delta z\}$,

$T^y_{i,k} = \{(x, y, z) \in D, (i-1)\Delta x \leq x \leq i\Delta x, (k-1)\Delta z \leq z \leq k\Delta z\}$,

$T^z_{i,j} = \{(x, y, z) \in D, (i-1)\Delta x \leq x \leq i\Delta x, (j-1)\Delta y \leq y \leq j\Delta y\}$,

for all $1 \leq i \leq N^x, 1 \leq j \leq N^y, 1 \leq k \leq N^z$, where $N^x, N^y$ and $N^z$ respectively denote the number of cells in the directions $x$, $y$ and $z$.

In every tube, a 1D optimal transportation is performed, and one can prove that, after dealing with all the particle-tubes in each direction, one obtains a uniform number of particles in every cell. It has to be noticed that the final transportation is clearly not optimal, and improvements of the above method, called Triangular Transportation Method, are to be done in subsequent studies.

4. modify the velocities from $\tilde{U}^{(n+1)}$ to $U^{(n+1)}$ in order to verify equation (2.4b), hence (2.3). This correction is a standard issue in CFD: the aim is to obtain a divergence free velocity field. This is done thanks to the resolution of the following Poisson problem:

$$\Delta P = \frac{1}{\Delta t} \nabla \cdot \langle \tilde{U}^{(n+1)} \rangle. \quad (2.8)$$

We supplement this equation with Neumann boundary conditions, and solve it thanks to solvers based on fast Fourier transforms (FFT). The equation above only concerns Eulerian unknowns, and hence does not contribute significantly to the numerical cost of the algorithm.

We then compute $U^{(n+1)}$ thanks to

$$U^{(n+1)} = \tilde{U}^{(n+1)} - \Delta t \nabla P. \quad (2.9)$$

The velocity field is then such that $\text{div}(U^{(n+1)}) = 0.$
3 Future works

In the coming months, we are planning to run MM5 with two different resolutions. The first mesh will be coarse, and will be devoted to feed our stochastic model. The second mesh will be finer, and we hope that our simulations will provide comparable results, with a better (smaller) computational cost (CPU time).

Then, we will use observations from measurements campaigns, namely the campaign FETCH that took place in 1998 in Southern France, for the validation of our simulations. We are going to use surface wind measurements as well as wind measurements from radio soundings from the ground up to several kilometers.

4 Acknowledgments

This work was partially supported by the French Agency for the Environment and Energy Management (ADEME). The authors also want to thank Eric Peirano for very helpful discussions related to this article.

Références