



On Fractional Gaussian Random Fields Simulations

Alexandre Brouste

Université du Maine

Jacques Istas

Université Pierre-Mendès-France

Sophie Lambert-Lacroix

Université Joseph Fourier

Abstract

To simulate Gaussian fields poses serious numerical problems: storage and computing time. The midpoint displacement method is often used for simulating the fractional Brownian fields because it is fast. We propose an effective and fast method, valid not only for fractional Brownian fields, but for any Gaussian fields. First, our method is compared with midpoint for fractional Brownian fields. Second, the performance of our method is illustrated by simulating several Gaussian fields. The software **FieldSim** is an R package developed in R and C and that implements the procedures on which this paper focuses.

Keywords: Gaussian fields, simulation, Hölder index.

1. Introduction

Rough phenomena arise in various fields (Frisch and Parisi 1985; Leland, Taqqu, Willinger, and Wilson 1994; Mandelbrot 1975; Peitgen and Saupe 1988; Pentland 1984): texture simulations and image processing, natural scenes (clouds, mountains) simulations, fluid mechanics, financial mathematics, ethernet traffic ... From a mathematical point of view, roughness is often measured by the Hölder index H . The lower H , the more rough the phenomenon, whereas H greater than 1 corresponds to smooth phenomenon. There is a real need to have models for rough phenomena. The fractional Brownian motions, in short FBM, introduced by Kolmogorov (1940) and further developed by Mandelbrot and Ness (1968), play a key role. FBM have then be extended in many directions: higher dimensions, anisotropy, multifractionality (Ayache, Léger, and Pontier 2002; Ayache and Lévy-Vehel 2000; Benassi, Bertrand, Cohen, and Istas 2000; Benassi, Cohen, and Istas 1998; Benassi, Jaffard, and Roux 1997; Bonami and Estrade 2004; Herbin 2006; Kamont 1996; Peltier and Levy-Véhel 1996).

The simulation of fractional Gaussian processes is not difficult in one dimension (e.g. the

surveys of Bardet, Lang, Oppenheim, Philippe, and Taqqu (2003); Coeurjolly (2000)). Let us recall the numerical complexity of some classical methods : the Cholesky method has a complexity of $O(N^3)$, where N is the size of the simulated sample path, Levinson's a complexity of $O(N^2 \log N)$ and Wood and Chan's a complexity of $O(N \log N)$. These complexities do not pose problems in dimension one, but become problematic in higher dimensions. These methods are indeed no longer tractable: these algorithms are time and memory expensive. Since its introduction by Lévy (1965) for Brownian motion, the random midpoint displacement method has been intensively used for generating fractional Brownian field (Fournier, Fussel, and Carpenter 1982; Peitgen and Saupe 1988; Voss 1985). This method is a rough approximation and is very fast.

Our approach is based on exact simulation plus a fast step, that is an improvement of the midpoint method. Let us be more precise. The aim is to simulate a Gaussian field over a fine grid. We first simulate the field in an exact way on a rough grid via the Cholesky method. We then propose a refinement of the midpoint method. The field is simulated, using a set of neighbors, and not only the nearest neighbors. This requires a computation of local weighting coefficients. This is a major difference with the midpoint method. For midpoint, these local coefficients are fixed to $1/4$. For our method, these local coefficients are exactly determined from the second-order structure. Contrary to the midpoint method, this allows us to simulate fields with arbitrary covariance as well rough as smooth.

One then needs to compare our methods called `fieldsim` with the midpoint method. Since the real aim is to simulate rough phenomenon, we estimate the Hölder index H of the simulation. The closer the estimation of H is, the better the simulation is. Our conclusion is the following. First, on fractional Brownian fields, `fieldsim` is slightly better than midpoint. Since the midpoint method is faster, this means that midpoint simulator is nevertheless relevant for fractional Brownian fields. Note that the Hölder index is only one criterion that can be used to assess the accuracy of the simulation. The procedure `fieldsim` produces fields with the desired Hölder index. But we do not focus on the difference between the simulated field covariance and the specified one. Second, we illustrate the good performances of our method by simulating several Gaussian fields, for which midpoint is no longer applicable.

The paper proceeds as follows. In the second section, we present our procedure `fieldsim`. Then we compare with the midpoint method. In the third section, we present our method on many examples. We recall in the appendix some classical results on the estimation of the Hölder index and our R package. Tables and figures are postponed at the end of the paper.

2. Method

After introducing some notations, we present the both steps of the procedure `fieldsim`: accurate and refined steps. Then we recall the random midpoint displacement method which is comparable with the procedure proposed here. We underline the drawbacks of this approach compared with ours.

2.1. Notations

Let d be a positive integer and $X(\cdot) = \{X(M), M \in \mathbb{R}^d\}$, be a real valued non stationary field with zero mean and second order moments. In this paper we are only concerned with the second order properties of the field $X(\cdot)$. It is convenient to use a geometrical approach

by considering the following Hilbert space \mathcal{M} , with the inner product $\langle U, V \rangle = E\{UV\} = Cov\{U, V\}$. The elements of \mathcal{M} are the linear combinations, with real coefficients, of elements of $\{X(M), M \in \mathbb{R}^d\}$ and their limits for mean square convergence. So the covariance function $R(\cdot, \cdot)$ is defined by:

$$R(M, M') = \langle X(M), X(M') \rangle = Cov\{X(M), X(M')\}, \quad (M, M') \in \mathbb{R}^d.$$

This function is nonnegative definite (n.n.d.), that is for all $n \geq 1$, for all real scalars $\lambda_1, \dots, \lambda_n$, and for all $M^1, \dots, M^n \in \mathbb{R}^d$,

$$\sum_{i,j=1}^n \lambda_i \lambda_j R(M^i, M^j) \geq 0.$$

Conversely, for any n.n.d. function $R(\cdot, \cdot)$, there exists an unique centered Gaussian field of second order structure given by $R(\cdot, \cdot)$.

2.2. The procedure `fieldsim`

Accurate simulation step

Let us recall that the goal of this paper is to give a procedure that yields discretization of sample path of the Gaussian field associated with any n.n.d. function $R(\cdot, \cdot)$. In the sequel, we denote by $X(\cdot)$ this sample path. Here we present the accurate simulation part of our procedure. Given a (regular) space discretization $\{M^i, i \in I\}$ of size n_I , the problem consists in giving a sample of a centered Gaussian vector of size n_I : $(X(M^i))_{i \in I}$ of covariance matrix \mathbf{R} given by $\mathbf{R}_{i,j} = R(M^i, M^j)$, $i, j \in I$. There exist many approaches to do that. We chose the procedure given by [Dégerine and Lambert-Lacroix \(2003\)](#) (see Theorem 2). This algorithm is based on Cholesky decomposition of the matrix \mathbf{R} in the sequential manner.

Refined simulation step

We need to introduce some additional notations. Let $X_{\mathcal{X}_I}(M)$, denote the orthogonal projection of $X(M)$ on the closed linear subspace $\mathcal{X}_I = \overline{\text{sp}}\{X(M^i), i \in I\}$, i.e. the linear predictor of $X(M)$ given $X(M^i), i \in I$. The partial innovation $X(M) - X_{\mathcal{X}_I}(M)$ is denoted by $\varepsilon_{\mathcal{X}_I}(M)$. Since $\varepsilon_{\mathcal{X}_I}(M)$ is uncorrelated with any variables of the space \mathcal{X}_I , we can obtain “accurate” simulation of $X(M)$ by $X_{\mathcal{X}_I}(M) + \sqrt{Var(\varepsilon_{\mathcal{X}_I}(M))}U$ where U is a centered and reduced Gaussian variable independent of $X(M^i), i \in I$. Notice that the coefficients weighting the variables $X(M^i), i \in I$ in $X_{\mathcal{X}_I}(M)$ and the variance of the partial innovation may be determined from the second order structure of the sequence $X(M^i), i \in I, X(M)$ (see [Dégerine and Lambert-Lacroix \(2003\)](#) for details). The drawback of this approach is when the simulated sequence size increases, we have to stock more and more quantities (filters of several partial innovation and associated variances) and to do more and more calculus. Even if that can be done in the case $d = 1$, it becomes numerically unfeasible when $d \geq 2$. To overcome this problem, a natural approach consists in replacing in the previous procedure the indexes set I by a set of indexes of neighbors of M . We denote by N_M this set. Notice that $X_{\mathcal{X}_{N_M}}(M)$ is the best linear combination of variables of \mathcal{X}_{N_M} approximating $X(M)$ in the sense that the variance of $X(M) - X_{\mathcal{X}_{N_M}}(M)$ is minimum. If we have to use only some variables of the set \mathcal{X}_{N_M} in order to obtain simulation of $X(M)$, the best way is to use $X_{\mathcal{X}_{N_M}}(M) + \sqrt{Var(\varepsilon_{\mathcal{X}_{N_M}}(M))}U$.

Let us remark that such a simulated process does not admit anymore $R(\cdot, \cdot)$ as a covariance function, but a covariance function that is a good approximation of $R(\cdot, \cdot)$.

2.3. Comparison with the random midpoint displacement method

The midpoint displacement method was developed for fractional Brownian field (see [Fournier et al. \(1982\)](#); [Peitgen and Saupe \(1988\)](#); [Voss \(1985\)](#)). For example in the scalar case ($d = 1$), the covariance of a standard fractional Brownian motion is given by

$$R(t, s) = \frac{1}{2} \{|t|^{2H} + |s|^{2H} - |t - s|^{2H}\},$$

with t and s in $[0, 1]$. The parameter H , which is usually called Hurst parameter, is a real in $(0, 1]$. For $H = 1/2$, we obtain the Brownian motion (with $R(t, s) = \inf(t, s)$). In the case $d > 1$, the absolute values in the definition of the covariance function below, are replaced by the Euclidean norm over \mathbb{R}^d .

This procedure is based on use of a recursive subdivision approach. For example in the scalar case, the procedure begins by assigning the null value to $X(0)$ and to $X(1)$. That means that one simulates more or less the bridge associated with X . To fairly compare the both approaches, we will generate $X(1)$ as a standard Gaussian variable instead of $X(1) = 0$. Then the segment $[0, 1]$ is divided in half to obtain one new grid point. The process at this new point is simulated as average of its two neighbors $X(0)$ and $X(1)$ plus a Gaussian variable with zero mean and appropriate variance. Subdivision continues to a desired level of recursion and the procedure is repeated for each new sub-grid. Precisely, for the iteration $n = 1, 2, \dots$, the process at the new grid points 2^{n-j} , $j = 2k + 1$, $0 \leq k \leq \frac{n-1}{2}$, is given by

$$X(2^{n-j}) = \frac{1}{2} \{X(2^{n-j+1}) + X(2^{n-j-1})\} + \sqrt{\frac{(1 - 2^{2H-2})}{2^{2nH}}} U.$$

Notice that $(1 - 2^{2H-2})/2^{2nH}$ is the variance of the variables $W(2^{n-j}) - \frac{1}{2}\{W(2^{n-j+1}) + W(2^{n-j-1})\}$, where $W(\cdot)$ is a fractional Brownian motion of Hurst parameter H .

When $d = 2$, one can use for example rectangular ground plane or triangular surfaces. For the first case, the procedure begins by assigning the null values to each of the four corners of the rectangular ground plane. Similarly to the scalar case, we generate some standard Gaussian variables for the three corners. Then the original boundaries of the ground plane are divided in half to obtain five new grid points. Simulations at these new grid positions are obtained as field averages at the two nearest neighbors of these points plus random value as for the scalar case. For the center point, one can use the field average at the four corners. The procedure is repeated for each new sub-grid. For the second case, two equilateral triangles are used, set side by side to form a parallelogram. At each level of recursion, the triangles are subdivided into successively smaller triangular surfaces.

The first difference between this approach and ours concerns the initialization since the procedure proposed here begins with accurate simulation step. Furthermore, the coefficients weighting the field at the two (resp. four) M neighbors in $X_{\mathcal{X}_I}(M)$ (where I is constituted of the M neighbors) are generally not equal to $1/2$ (resp. $1/4$). Precisely, in the case $d = 1$ and $H = 1/2$, let us consider t , s and u such that $0 < s < u < t \leq 1$. We obtain $X_{\mathcal{X}_{\{s,t\}}}(u) = 1/2(X(s) + X(t))$ and $\varepsilon_{\mathcal{X}_{\{s,t\}}}(u) = \varepsilon_{\mathcal{X}_I}(u)$ for any I that contains s and t and does not contain any points in $]s, t[$. So we need only two neighbors in order to simulate the

process at u . On the other hand, if $s = 0$ then $X_{\mathcal{X}_{\{s,t\}}}(u) = uX(t)$. If we take in our procedure only the point 0 and 1 for the accurate step and use after two neighbors in the refined step, we obtain a procedure to simulate the process in a accurate way. That is not the case for the random midpoint displacement method. Indeed each time the zero point is concerned as neighbor of u , the coefficient weighted the process at the neighbor on the left-hand side is equal to $1/2$ instead of u . Otherwise when $d \neq 1$ or $H \neq 1/2$, the coefficients used in the projection depend on the position and are not equal to $1/2$ (resp. $1/4$) when two (resp. four) neighbors are concerned. So the random midpoint displacement method does not use all the information contained in the neighbors. However, the both approaches to simulate fractional Brownian fields (see Section 3.1) leads to comparable results; but the procedure `fieldsim` can be applied for any Gaussian random field, provided that its covariance function is known.

3. Numerical results

In this section, we illustrate the method proposed here through simulations for several classes of fields.

3.1. Fractional Brownian fields

The standard fractional Brownian fields are defined through their covariance function (e.g. [Samorodnitsky and Taqqu \(1994\)](#)):

$$R(M, M') = \frac{1}{2} \{ \|M\|^{2H} + \|M'\|^{2H} - \|M - M'\|^{2H} \},$$

where the Hurst parameter H is real in $(0, 1]$. The case $H = 1$ is degenerated and will be omitted in the following. For various size of sample $(N + 1)^2$ ($N = 64, 128, 256$ and 512) and various values of H ($H = 0.1, 0.3, 0.5, 0.7$ and 0.9), we generate 100 paths of fractional Brownian fields, discretized uniformly on $[0, 1]^2$. We use the both procedures `midpoint` and `fieldsim`. For all the simulations generated by `fieldsim`, we have chosen in the accurate simulation step, a regular space discretization of size 25 and in the refined simulation step a number of neighbors equal to 8. Figure 1 summarizes typical paths for $N = 64$, $H = 0.1, 0.5$ and 0.9 . The lower H , the more irregular is the field. For each path, the parameter H is estimated via the generalized quadratic variations introduced in [Istas and Lang \(1997\)](#). We denote by \hat{H}_N this estimator. Some Boxplots (see Figure 2) illustrate the results. In order to explore the quality of these both simulation methods, we propose to use the same approach as in [Coeurjolly \(2000\)](#) consisting in testing the value of H . Precisely, we have the following asymptotic normality result:

$$N^{\frac{d}{2}}(\hat{H}_N - H) \xrightarrow{\mathcal{D}} \mathcal{N}(0, \gamma_H^2),$$

where the convergence is in distribution and γ_H^2 is some constant. Some details such as the expression of \hat{H}_N and the one of γ_H^2 are given in the Appendix. Table 1 gives the constant γ_H^2 for different values of H .

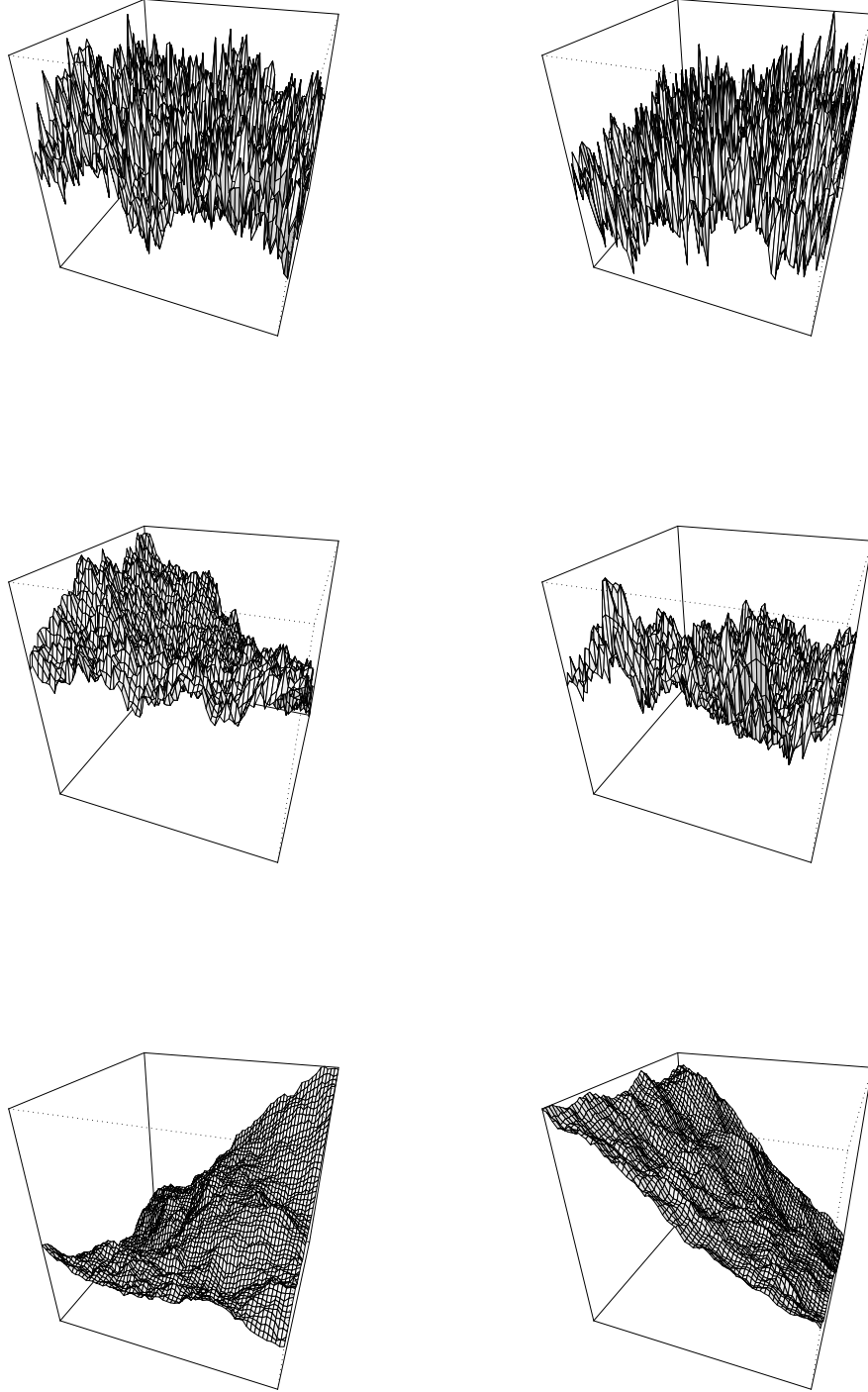


Figure 1: Typical paths of fractional Brownian fields (for $N = 64$, $H = 0.1$ (top), 0.5 (middle) and 0.9 (bottom)) simulated by `midpoint` or `fieldsim`.

H	0.1	0.3	0.5	0.7	0.9
γ_H^2	17.91	16.82	15.66	14.45	13.20

Table 1: Constant γ_H^2 for different values of H .

We use this property to extract an efficient method to compare the both procedures. For different values of H and N , we first simulate 100 paths of a fractional Brownian field. For the i th path, we compute the estimation of the Hurst parameter denoted by \hat{h}_N^i . Then the percentage test success is estimated by

$$S_H = \frac{1}{100} \sum_{i=1}^{100} 1_{\hat{h}_N^i \in]H - u_{0.05} \frac{\gamma_H}{N}, H + u_{0.05} \frac{\gamma_H}{N}[},$$

where $u_{0.05} \approx 1.96$ is the 95th percentile of the standard Gaussian distribution. The results are given in Table 2.

H	0.1	0.3	0.5	0.7	0.9
midpoint					
$N = 64$	91%	91%	91%	90%	87%
$N = 128$	89%	89%	87%	85%	84%
$N = 256$	95%	95%	94%	93%	92%
$N = 512$	90%	85%	84%	83%	83%
fieldsim					
$N = 64$	86%	89%	89%	89%	90%
$N = 128$	86%	88%	91%	89%	89%
$N = 256$	93%	96%	94%	94%	95%
$N = 512$	88%	88%	87%	87%	88%

Table 2: Percentage test success for **midpoint** and **fieldsim**.

The both methods work well since the level 95% is almost always reached. One can notice that **fieldsim** is slightly better than **midpoint**. However, considering the computing times (see Table 3), it seems preferable to use **midpoint** to simulate fractional Brownian fields.

N	64	128	256	512
midpoint	1.20	4.93	19.35	76.32
fieldsim	23.83	105.89	426.47	1673.85

Table 3: Mean CPU time for **midpoint** and **fieldsim**.

3.2. Multifractional Brownian fields

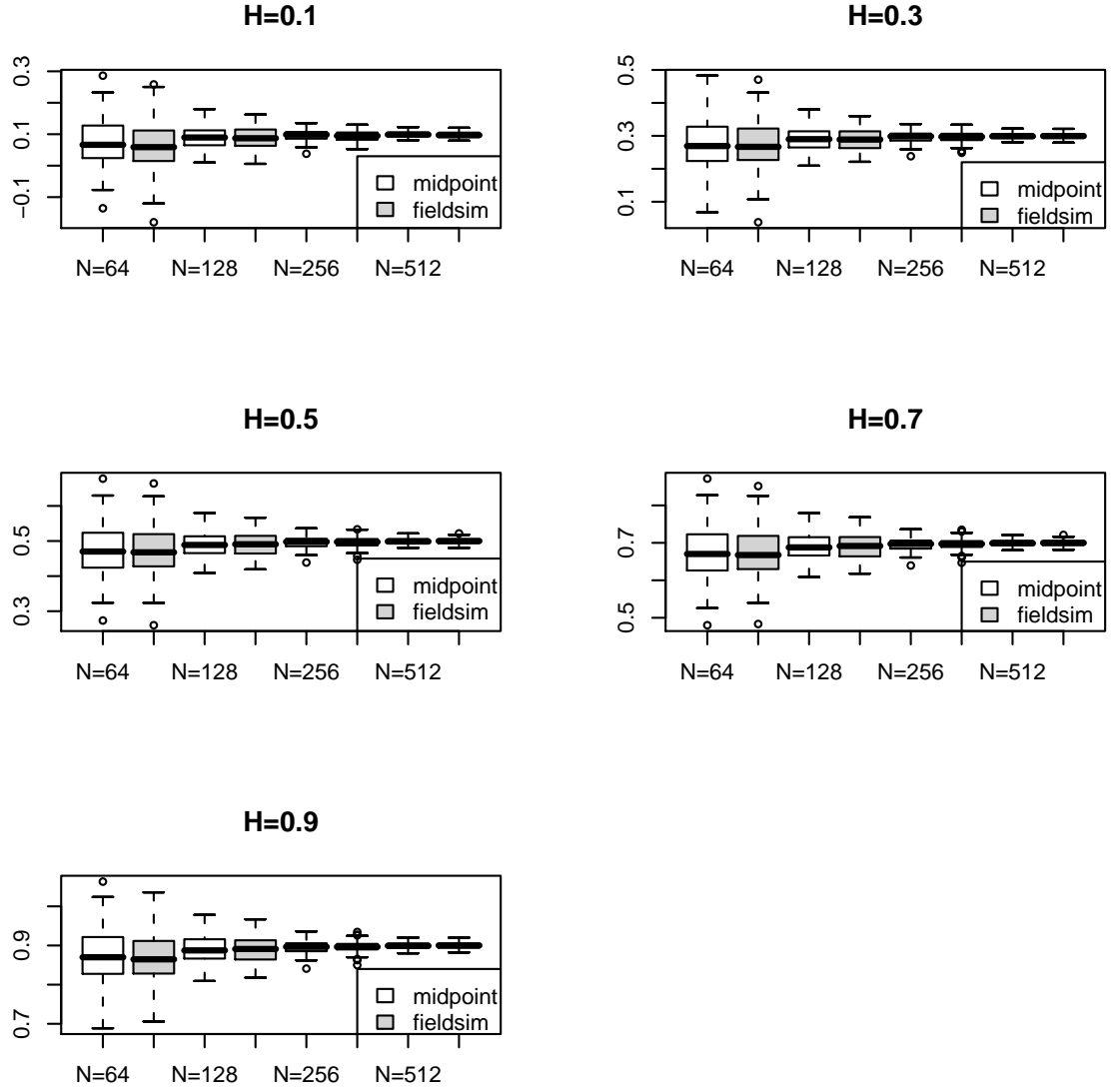


Figure 2: Boxplots of estimators of H for different values of H and N for 100 paths simulated by midpoint or fieldsim.

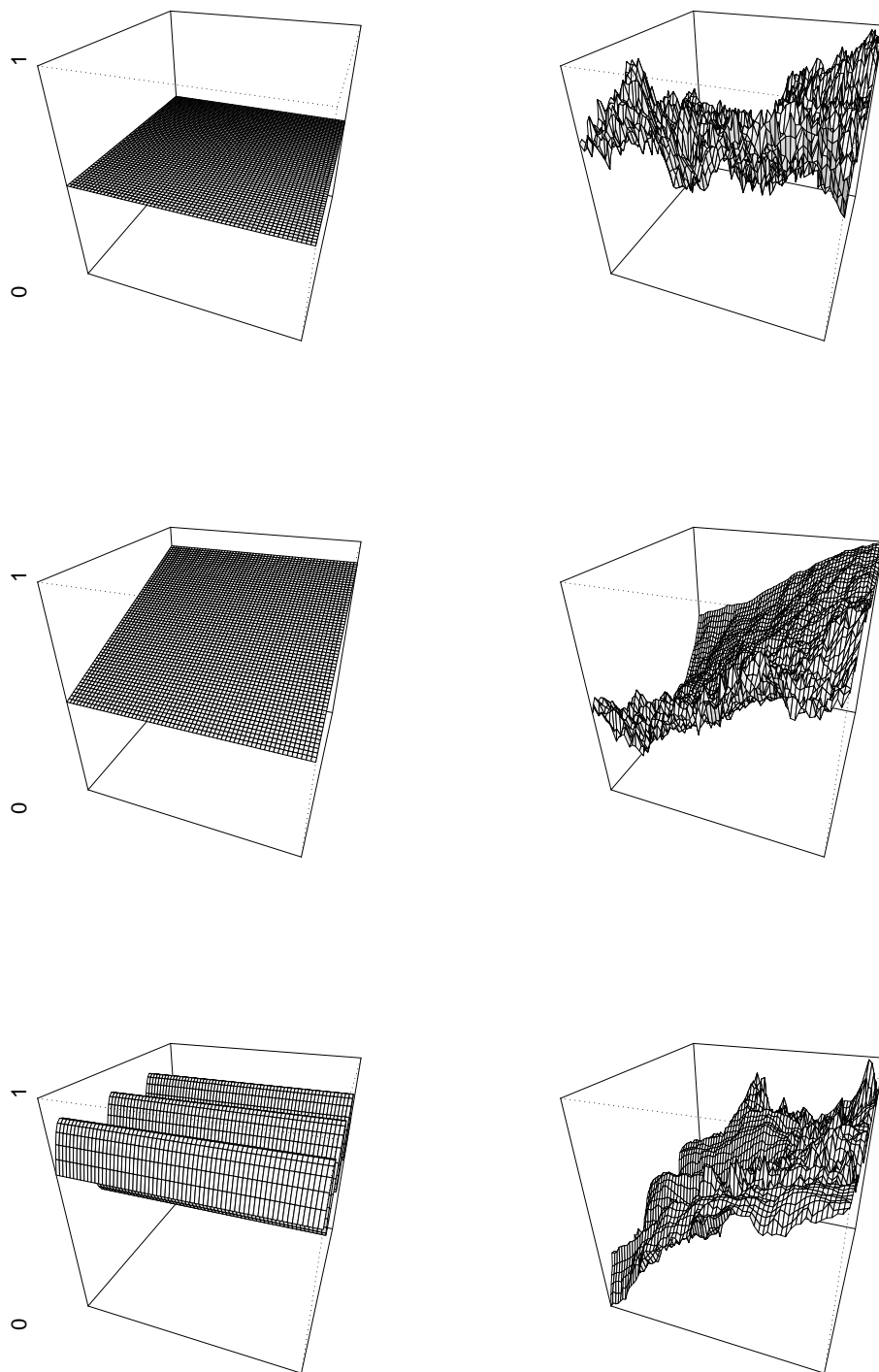


Figure 3: Typical paths of multifractional Brownian fields (on the right) for $N = 64$ and Hurst functions (on the left) for $H(t) = 0.5$ (top), $H(t) = 0.5 + 0.4t_1$ (middle) and $H(t) = 0.7 + 0.2 \sin(2\pi t_1)$ (bottom) simulated by `fieldsim`.

The standard multifractional Brownian fields are defined through their covariance function (see (Peltier and Levy-Véhel 1996; Benassi *et al.* 1997)):

$$R(M, M') = D(M, M') \left\{ \|M\|^{H(M)+H(M')} + \|M'\|^{H(M)+H(M')} - \|M - M'\|^{H(M)+H(M')} \right\},$$

where

$$D(M, M') = \frac{C\left(\frac{H(M)+H(M')}{2}\right)^2}{2C(H(M))C(H(M'))},$$

and

$$C(h) = \left(\frac{\pi^{\frac{d+1}{2}} \Gamma\left(h + \frac{1}{2}\right)}{h \sin(\pi h) \Gamma(2h) \Gamma\left(h + \frac{d}{2}\right)} \right)^{\frac{1}{2}}$$

and the Hurst parameter is a continuous function $H : \mathbb{R}^d \longrightarrow (0, 1)$.

Different paths for size sample $N = 65^2$ and Hurst function $H_0(M) = 0.5$, $H_1(M) = 0.5 + 0.4M_1$ et $H_2(M) = 0.7 + 0.2 \sin(6\pi M_1)$, where $M = (M_1, M_2) \in [0, 1]^2$, are summarized in Figure 3. When the function $H(M)$ is constant as for $H_0(M)$, one recognizes the fractional Brownian fields described in Section 3.1. To compare with precedent section, CPU times for fields of 1025×1025 points are around 1400 for fractional Brownian Fields and 3200 for multifractional Brownian Fields.

In order to illustrate the quality of `fieldsim`, we propose to estimate from different sample paths, the function $H(M)$ over a regular grid of $[0, 1]^2$ and to compare it with $H(M)$. To do this, we use the procedure described in Lacaux (2004) based on localized quadratic variations. We first simulate 100 paths of size 257^2 for the both functions $H_1(M)$ and $H_2(M)$. Here we have chosen in the accurate simulation step, a regular space discretization of size 9 and in the refined simulation step a number of neighbors equal to 4. The functions $H_1(M)$ and $H_2(M)$ are estimated on a regular space discretization of size 63^2 using a bandwidth equal to 0.125 for $H_1(M)$ and 0.078 for $H_2(M)$. In Figure 4, we plot the average (over the 100 paths) estimated Hurst function for the both functions $H_1(M)$ and $H_2(M)$. One can see that `fieldsim` works well.

3.3. Two parameters fractional Brownian fields

The standard bi-fractional Brownian fields are defined through their covariance function (see Houdré and Villa (2003)):

$$R(M, M') = \frac{1}{2} \left\{ (\|M\|^{2H} + \|M'\|^{2H})^K - \|M - M'\|^{2HK} \right\},$$

where the Hurst parameter H is real in $(0, 1)$ and K in $(0, 1]$. For $K = 1$ the corresponding process is a standard fractional Brownian field of Hurst parameter H , but, when $K < 1$, increments of the process start to be non stationary but remain locally self-similar of order HK .

Different paths for size sample $N = 64$ and parameters $H = 0.5$, $K = 1$ and $H = 0.9$, $K = 0.55$ are plotted on Figure 5.

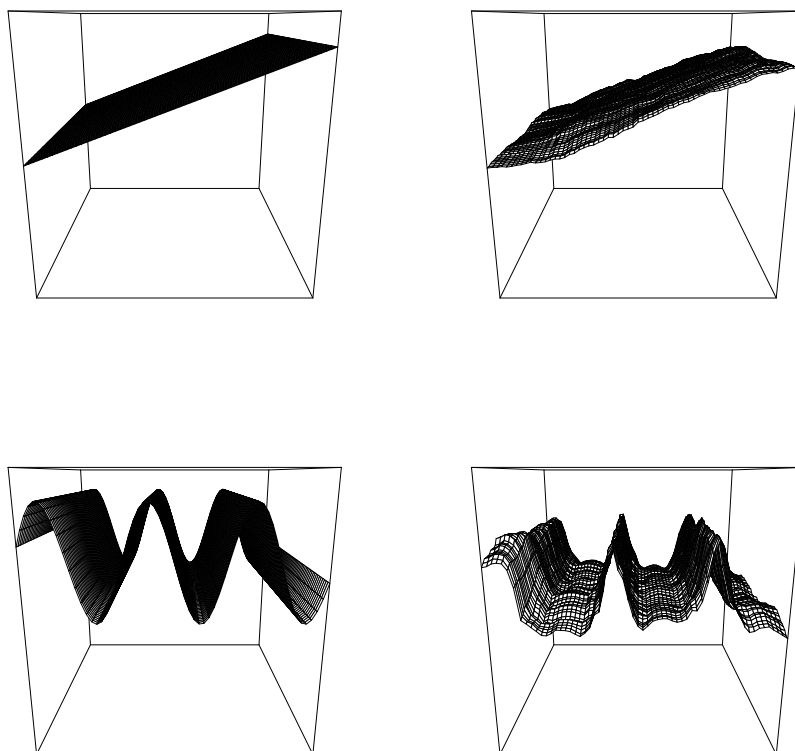


Figure 4: True Hurst function (on left) and average estimated Hurst function with localized quadratic variations method (on right) for multifractional Brownian fields with $H(t) = 0.5 + 0.4t_1$ (on top) and $H(t) = 0.7 + 0.2 \sin(2\pi t_1)$ (on bottom).

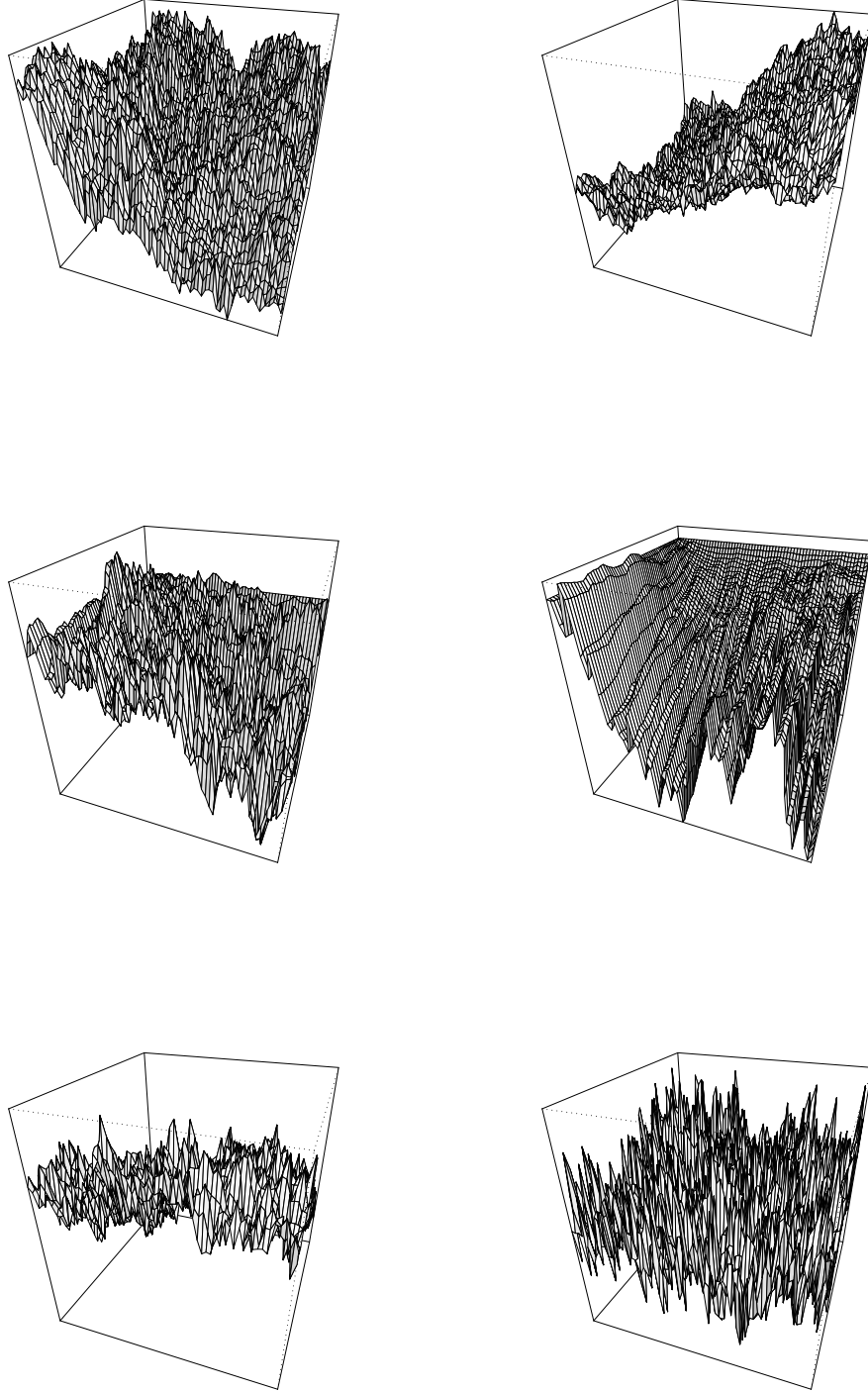


Figure 5: Typical paths of two parameters fractional Brownian fields (on top, for $N = 64$, $H = 0.5$, $K = 1$ (left) and $H = 0.9$, $K = 0.55$ (right)), fractional Brownian sheets (in the middle, for $N = 64$, $H_1 = 0.5$ and $H_2 = 0.5$ (left) and $H_1 = 0.9$, $H_2 = 0.55$ (right)) and hyperbolic fractional Brownian field (on bottom left for $H = 0.1$ and right for $H = 0.1$ on $[-\frac{1}{2}, \frac{1}{2}]^2$) simulated by `fieldsim`.

3.4. Fractional Brownian sheets

The standard fractional Brownian sheets are defined through their covariance function (see [Kamont \(1996\)](#)):

$$R(M, M') = \frac{1}{2^d} \prod_{i=1}^d \left\{ |M_i|^{2H_i} + |M'_i|^{2H_i} - |M_i - M'_i|^{2H_i} \right\},$$

where $M = (M_1, M_2, \dots, M_d)$, $M' = (M'_1, M'_2, \dots, M'_d)$ are in \mathbb{R}^d and $H = (H_1, H_2, \dots, H_d)$ stands for the multivariate Hurst index in \mathbb{R}^d , $0 < H_i < 1$.

Fractional Brownian sheets do not have stationary increments but have stationary increments with respect to each variable. Therefore they are anisotropic fields but they are self-similar of index $\prod_{i=1}^d H_i$. Typical paths of corresponding fields of parameters $H_1 = 0.5$, $H_2 = 0.5$ and $H_1 = 0.5$, $H_2 = 0.9$ are also plotted on Figure 5.

3.5. Space-Time deformed fractional Brownian fields

The space-time fractional Brownian fields are defined through their covariance function (see [Bégyn \(2006\)](#)):

$$R(M, M') = \frac{\sigma(M)\sigma(M')}{2} \left\{ \|\tau(M)\|^{2H} + \|\tau(M')\|^{2H} - \|\tau(M) - \tau(M')\|^{2H} \right\},$$

where the Hurst parameter H is real in $(0, 1)$, $\sigma : \mathbb{R}^d \rightarrow \mathbb{R}_*$ is a Σ -Hölder continuous function of order $\Sigma > H$ and $\tau : \mathbb{R}^d \rightarrow \mathbb{R}^d$ is a continuously differentiable function such that $\nabla \tau(M) \neq 0$ for all $M \in \mathbb{R}^d$.

Path of corresponding field with $d = 2$, for $H = 0.7$ and of functional parameters $\sigma(M) = e^{-(M_1+M_2)}$, $\tau(M) = \left(e^{\frac{M_1}{H}}, e^{\frac{M_2}{H}}\right)$ is finally plotted on Figure 6.

3.6. Hyperbolic fractional Brownian fields

Let $D_d = \{M \in \mathbb{R}^d, \|M\| < 1\}$, where $\|\cdot\|$ is the usual Euclidean norm. For $M, M' \in D$, define

$$\delta(M, M') = 2 \frac{\|MM'\|}{(1 - \|M\|^2)(1 - \|M'\|^2)},$$

and

$$\rho(M, M') = \operatorname{arccosh}(1 + \delta(M, M')).$$

The metric space (D_d, ρ) is a model of hyperbolic space (e.g. [Helgason \(1962\)](#)). When $d = 2$, (D_2, ρ) is the Poincaré's disk. The Hyperbolic fractional Brownian field (in short HFBBF) is the centered Gaussian field with covariance function

$$R(M, M') = \frac{1}{2} (\rho^{2H}(O, M) + \rho^{2H}(O, M') - \rho^{2H}(M, M')),$$

where O is the origin of \mathbb{R}^d . The HFBBM exists if and only if $0 < H \leq 1/2$ ([Istas \(2005\)](#)). Paths of HFBBF, with $d = 2$, $H = 0.3$ and $H = 0.5$ are plotted in Figures 5 and 6.

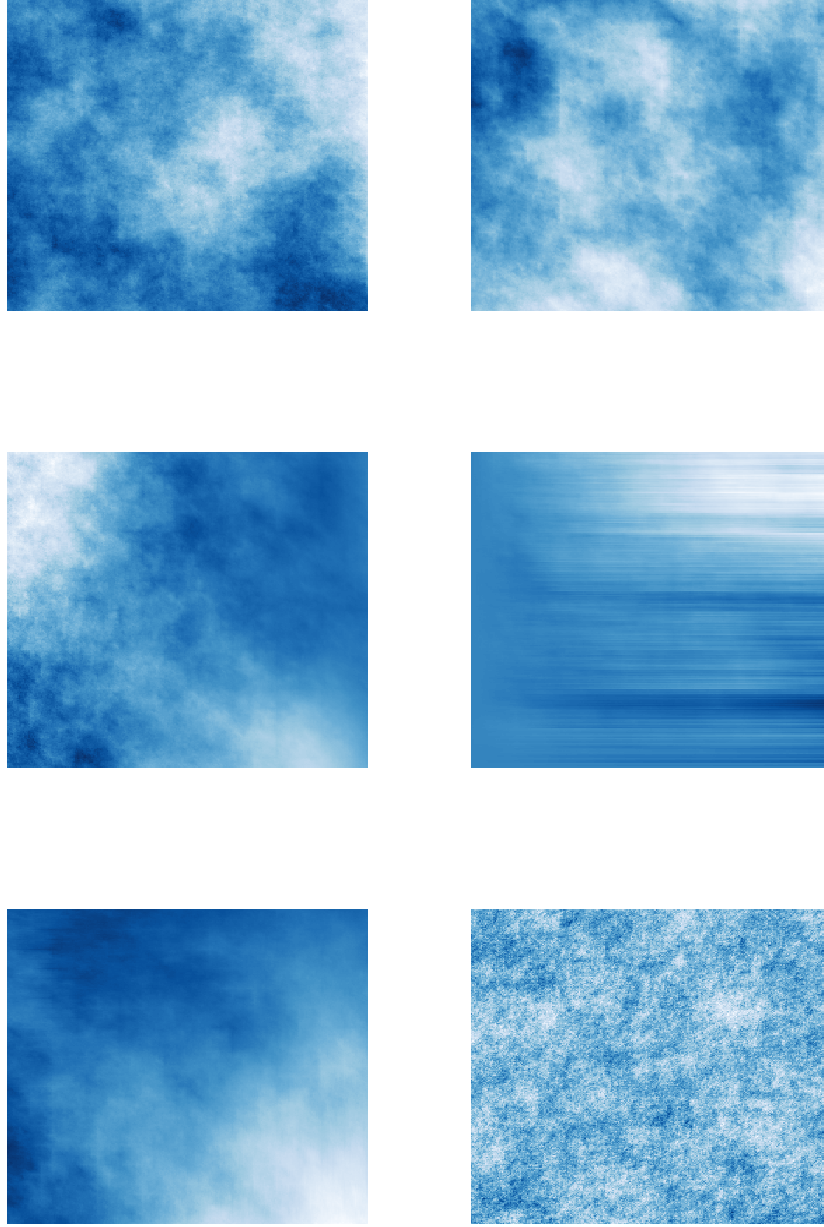


Figure 6: 256×256 – Fractional Brownian fields (on top, for $H = 0.5$ on the left and $H = 0.7$ on the right), multifractional Brownian field (middle left for $H(t) = 0.5 + 0.4t_1$), fractional Brownian sheet (middle right for $H_1 = 0.5$, $H_2 = 0.9$), space-time deformed Brownian field (bottom left with $\sigma(t) = e^{-(t_1+t_2)}$ and $\tau = e^{\frac{1}{H}t}$) and hyperbolic fractional Brownian fields (bottom right for $H = 0.1$ on $[-\frac{1}{2}, \frac{1}{2}]^2$) simulated by **fieldsim**.

4. Appendix

4.1. Asymptotically normality of the Hurst parameter estimator

Let $X(\cdot)$ be a fractional Brownian field with Hurst parameter H . Let us suppose that one has a sample path of this process over the grid $(k, l)^T/N$, $k, l = 0, \dots, N$. We consider the generalized variations based on the discrete second derivatives,

$$V_N(X) = \sum_{k=1}^{N-1} \sum_{l=1}^{N-1} \left(\sum_{i=-1}^1 \sum_{j=-1}^1 a_i a_j X \left(\frac{k+i}{N}, \frac{l+j}{N} \right) \right)^2,$$

where $a_{-1} = a_1 = 1$ and $a_0 = -2$. The associated estimator of the Hurst parameter is defined by (see [Istas and Lang \(1997\)](#)),

$$\hat{H}_N = \frac{1}{2} \log_2 \left(\frac{V_{N/2}(X)}{V_N(X)} \right) + 1.$$

Without restriction, one can consider only even N , so \hat{H}_N is well defined. It can be shown (see [Biermé \(2005\)](#)) that

$$N(\hat{H}_N - H) \xrightarrow{\mathcal{D}} \mathcal{N}(0, \gamma_H^2),$$

where the convergence is in distribution. The constant γ_H^2 is given by:

$$\gamma_H^2 = \frac{1}{(C_1 \log 2)^2} \left(\frac{5}{4} C_2 - 2^{-2H+1} C_3 \right),$$

where C_1 , C_2 and C_3 are positive constants given by

$$\lim_{N \rightarrow +\infty} N^{2H-2} \mathbb{E}(V_N(X)) = C_1, \quad \lim_{N \rightarrow +\infty} N^{4H-2} \text{Var}(V_N(X)) = C_2,$$

and

$$\lim_{N \rightarrow +\infty} N^{4H-2} \text{Cov}(V_N(X), V_{N/2}(X)) = C_3.$$

Constants C_1 , C_2 and C_3 are computed in [Biermé \(2005\)](#) in a general setting. Let us give them in our case in a more tractable way:

$$C_1 = -\frac{1}{2} u_0^0, \tag{1}$$

$$C_2 = \frac{1}{2} \left((u_0^0)^2 + 4 \sum_{l=1}^{\infty} \sum_{k=0}^{\infty} (u_l^k)^2 \right), \tag{2}$$

$$C_3 = \frac{1}{8} \left((\tilde{u}_0^0)^2 + \sum_{l=1}^{\infty} \sum_{k=0}^{\infty} \left\{ (\tilde{u}_l^k)^2 + (\tilde{u}_{-l}^k)^2 + (\tilde{u}_l^{-k})^2 + (\tilde{u}_{-l}^{-k})^2 \right\} \right), \tag{3}$$

where

$$\begin{aligned} u_l^k &= \sum_{i,j,i',j'=-1}^1 a_{i,j}^{i',j'} ((k+i-i')^2 + (l+j-j')^2)^H, \\ \tilde{u}_l^k &= \sum_{i,j,i',j'=-1}^1 a_{i,j}^{i',j'} ((k+2i-i')^2 + (l+2j-j')^2)^H, \end{aligned}$$

where $a_{i,j}^{i',j'} = a_i a_j a_{i'} a_{j'}$. Indeed, we have

$$\begin{aligned} \mathbb{E}(V_N(X)) &= \sum_{i,j,i',j'=-1}^1 a_{i,j}^{i',j'} \mathbb{E} \left(X \left(\frac{k+i}{l+j} \right) X \left(\frac{k+i'}{l+j'} \right) \right), \\ &= \frac{1}{2} N^{-2H} \sum_{i,j,i',j'=-1}^1 a_{i,j}^{i',j'} \left(\left\| \frac{k+i}{l+j} \right\|^{2H} + \left\| \frac{k+i'}{l+j'} \right\|^{2H} - \left\| \frac{i-i'}{j-j'} \right\|^{2H} \right). \end{aligned}$$

Since $\sum_{i=-1}^1 a_i = 0$,

$$\mathbb{E}(V_N(X)) = -\frac{1}{2} (N-1)^2 N^{-2H} \sum_{i=-1}^1 \sum_{j=-1}^1 \sum_{i'=-1}^1 \sum_{j'=-1}^1 \left\| \frac{i-i'}{j-j'} \right\|^{2H},$$

that leads to the expression (1) for C_1 .

Concerning the relationship for C_2 , let us recall that for any centered Gaussian variables X and Y , we have,

$$\mathbb{E}(X^2 Y^2) - \mathbb{E}(X^2) \mathbb{E}(Y^2) = 2(\mathbb{E}(XY))^2.$$

That leads to

$$\begin{aligned} \text{Var}(V_N(X)) &= 2 \sum_{k,l,k',l'=1}^{N-1} \left\{ \mathbb{E} \left\{ \left(\sum_{i,j=-1}^1 a_i a_j X \left(\frac{k+i}{l+j} \right) \right) \right. \right. \\ &\quad \left. \left(\sum_{i',j'=-1}^1 a_{i'} a_{j'} X \left(\frac{k'+i'}{l'+j'} \right) \right) \right\} \right\}^2 \\ &= \frac{1}{2} N^{-4H} \sum_{k,l,k',l'=1}^{N-1} (\tilde{u}_{l-l'}^{k-k'})^2, \end{aligned}$$

because of $\sum_{i=-1}^1 a_i = 0$. Changing variables $k - k'$ into k'' and $l - l'$ into l'' provides,

$$\text{Var}(V_N(X)) = \frac{1}{2} N^{-4H} (N-1)^2 \left((u_0^0)^2 + 4 \sum_{l''=1}^{\infty} \sum_{k''=0}^{\infty} \left(1 - \frac{l''}{N-1} \right) \left(1 - \frac{k''}{N-1} \right) (u_{l''}^{k''})^2 \right),$$

what leads to the result provided that the sequence of terms $(u_l^k)^2$ converges. To prove that, we can express u_l^k as follows,

$$u_l^k = (k^2 + l^2)^H \sum_{i,j,i',j'=-1}^1 a_{i,j}^{i',j'} \left(1 + \epsilon_{i,j}^{i',j'}(k, l) \right)^H,$$

where

$$\epsilon_{i,j}^{i',j'}(k,l) = \frac{(i-i')^2}{k^2+l^2} + \frac{(j-j')^2}{k^2+l^2} + \frac{2(i-i')k}{k^2+l^2} + \frac{2(j-j')l}{k^2+l^2}.$$

Since $\sum_{i=-1}^1 a_i = \sum_{i=-1}^1 i a_i = 0$,

$$\sum_{i,j,i',j'=-1}^1 a_{i,j}^{i',j'} (i-i')^n (j-j')^m = 0, \quad n, m < 4,$$

and using Taylor's expansion of the function $f(x) = (1+x)^H$ up to order 4, we obtain

$$u_l^k = 4(k^2+l^2)^{H-4} \sum_{i,j,i',j'=-1}^1 a_{i,j}^{i',j'} f^{(4)}(\theta_{i,j}^{i',j'}(k,l)) (i-i')^4 (j-j')^4,$$

where $\theta_{i,j}^{i',j'} \in [0, \epsilon_{i,j}^{i',j'}(k,l)]$. On the other hand, for large enough values of k or l , $|\epsilon_{i,j}^{i',j'}(k,l)|$ can be bounded by 1 and then there exists some constant C such that $(u_l^k)^2 \leq C(k^2+l^2)^{2H-8}$. Since $H \in (0, 1]$, the series of terms $(k^2+l^2)^{2H-8}$ (consequently $(u_l^k)^2$) converges.

The expression given for C_3 is obtained in a similar way.

4.2. Using FieldSim

Implementation of FieldSim

FieldSim is a set of R functions that allows performing simulations of Gaussian Fractional Fields with known covariance function. Three classes of functions are implemented:

- Simulation functions `midpoint` and `fieldsim` that perform simulations of the path of the process. Procedure `fieldsim` uses Levinson-Durbin algorithm, thanks to the covariance function and the number of accurate and refined steps of the algorithm, parameters set by the user. `C2D` function has been implemented for the covariance function of multifractional Brownian Fields.
- Estimation functions: `quadvar` yields estimation of the Hurst parameter of a fractional Brownian field using the quadratic variations method [Istas and Lang \(1997\)](#). `locquadvar` performs estimation of the multifractional function at a given point for a multifractional Brownian field using the procedure described in [Lacaux \(2004\)](#). `H.test` gives results relating to the test presented at Subsection [3.1](#).
- C subroutine `vf` performs the tasks that are consuming because of the number of loops. `gamma2` and `quadvaraux` are other internal functions.

The R environment is the only user interface. Function `fieldsim` calls the C subroutine `vf` whose result is returned to R.

In order to make it easier for the reader not used to R language, we detail the call to functions and the commands used to produce graphical outputs.

Fractional Brownian Fields

To simulate Fractional Brownian Fields (see Section [3.1](#)), one needs to specify the Hurst parameter $0 < H < 1$ and the covariance function $R(M, M')$. For instance,

```

R> set.seed(626)
R> R <- function(x, H = 0.9) {
+   1/2*((x[1]^2 + x[2]^2)^H + (x[3]^2 + x[4]^2)^H -
+   ((x[1] - x[3])^2 + (x[2] - x[4])^2)^H)
+ }
R> FBF09 <- fieldsim(R, Eleve1 = 2, Rlevel = 4, nbNeighbor = 8)

```

simulates the corresponding process, with `Eleve1` accurate simulation steps and `Rlevel` refined simulation steps with `nbneighbor` neighbors. That corresponds to a sample path of size $(2^{\text{Eleve1}+\text{Rlevel}} + 1)^2$ with a grid of size $(2^{\text{Eleve1}} + 1)^2$ for the accurate simulation steps.

The result of the function `fieldsim` is a R Object of class `list`. It contains the following elements:

- vectors `Zrow`, `Zcol` of x and y coordinates and matrix Z of the simulated path of the process;
- real `time` that gives the CPU time.

Thus to produce the graph on the Figure 1, one have to call the function `persp` as

```

R> x <- FBF09$Zrow
R> y <- FBF09$Zcol
R> z <- FBF09$Z
R> persp(x, y, z, phi = 30, theta = 110, shade = 0.1, axes = FALSE)

```

Other illustrations can be done (Figure 6 for fractional and multifractional Brownian Fields), for instance *clouds* representation.

```

R> library(RColorBrewer)
R> colramp <- colorRampPalette(brewer.pal(9, "Blues"))
R> image(z, col = colramp(128))

```

To estimate the Hurst index in the Fractional Brownian case, one uses the implemented estimation function `quadvar` as

```
R> quadvar(Z = FBF09$Z)
```

```
[1] 0.828395
```

One can now compare quadratic variations estimation with the true value of H . To do this, one can use the procedure `H.test` in the following way

```
R> H.test(Z = FBF09$Z, H = 0.9, alternative = "two.sided", conf.level = 0.95)
```

Multifractional Brownian Fields

To simulate Multifractional Brownian Fields, one needs to specify the function $H(M)$, $M \in \mathbb{R}^2$, and the covariance function $R(M, M')$. Function $C(H)$, $H \in (0, 1[$, (see Section 3.2) have been implemented and can be called by `C2D`. For instance,

```

R> F <- function(y){0.4*y + 0.5}
R> R<-function(x, Fun = F){
+   H1 <- Fun(x[1])
+   H2 <- Fun(x[3])
+   alpha <- 1/2*(H1 + H2)
+   C2D(alpha)^2/(2*C2D(H1)*C2D(H2))*((x[1]^2 + x[2]^2)^alpha +
+   (x[3]^2 + x[4]^2)^alpha - ((x[1] - x[3])^2 + (x[2] - x[4])^2)^alpha)
+ }
R> MBFaff <- fieldsim(R, Elevel = 1, Rlevel = 7, nbNeighbor = 4)

```

simulates the corresponding path.

Thus to produce the graph such as Figure 3, one calls the function `persp` as

```

R> x <- MBFaff$Zrow
R> y <- MBFaff$Zcol
R> z <- MBFaff$Z
R> persp(x, y, z, phi = 30, theta = 110, shade = 0.1, axes = FALSE)

```

To estimate the function $H(M)$ at the point $M = (0.5, 0.5)^T$, one uses the following command

```

R> locquadvar(MBFaff$Z, t = c(0.5, 0.5), h = 0.125)

```

Two Parameters Fractional Brownian Fields

To simulate Two Parameters Fractional Brownian Fields (see Figure 5), one needs to specify the parameters $0 < H < 1$, $0 < K \leq 1$ and the covariance function (see Section 3.3). For instance,

```

R> R <- function(x, H = 0.5, K = 1){
+   1/2^K*((x[1]^2 + x[2]^2)^H + (x[3]^2 + x[4]^2)^H)^K -
+   ((x[1] - x[3])^2 + (x[2] - x[4])^2)^(H*K))
+ }
R> 2pFBB <- fieldsim(R, Elevel = 1, Rlevel = 5, nbNeighbor = 4)

```

Fractional Brownian Sheets

To simulate Fractional Brownian Sheet (see Figure 5), one indicates the multivariate Hurst index $H = (H_1, H_2) \in (0, 1)^2$ and the covariance function (see Section 3.4). For instance,

```

R> R <- function(x, H1 = 0.9, H2 = 0.5){
+   1/4*( abs(x[1])^(2*H1) + abs(x[3])^(2*H1) - abs(x[1] - x[3])^(2*H1))
+   *(abs(x[2])^(2*H2) + abs(x[4])^(2*H2) - abs(x[2] - x[4])^(2*H2))
+ }
R> FBS <- fieldsim(R, Elevel = 1, Rlevel = 5, nbNeighbor = 4)

```

Space-Time deformed Fractional Brownian Fields

To simulate Space-time deformed Fractional Brownian Fields (see Figure 5), one has to specify the Hurst parameter $0 < H < 1$, functional parameters τ and σ and the covariance function (see Section 3.5). For instance

```

R> tau <- function(y, H = 0.7){exp(y/H)}
R> R <- function(x, Fun = tau, H = 0.7){
+   1/2*((Fun(x[1])^2 + Fun(x[2])^2)^H
+     + (Fun(x[3])^2 + Fun(x[4])^2)^H
+     - ((Fun(x[1]) - Fun(x[3]))^2
+       + (Fun(x[2]) - Fun(x[4]))^2)^H)
+ }
R> res <- fieldsim(R, Elelevel = 1, Rlevel = 5, nbNeighbor = 4)

```

Hyperbolic fractional Brownian Fields

To simulate Hyperbolic fractional Brownian Fields (see Figure 6) on $[-\frac{1}{2}, \frac{1}{2}]$, one has to specify the Hurst parameter $0 < H \leq \frac{1}{2}$ and the specific covariance function (see Section 3.6). For instance

```

R> R <- function(x, H = 0.1){
+   1/2*(acosh(1 + 2*((x[1] - 1/2)^2 + (x[2] - 1/2)^2)/
+     (1 - ((x[1] - 1/2)^2 + (x[2] - 1/2)^2)))^(2*H) +
+   acosh(1 + 2*((x[3] - 1/2)^2 + (x[4] - 1/2)^2)/
+     (1 - ((x[3] - 1/2)^2 + (x[4] - 1/2)^2)))^(2*H) -
+   acosh(1 + 2*((x[1] - x[3])^2 + (x[2] - x[4])^2)/
+     ((1 - ((x[1] - 1/2)^2 + (x[2] - 1/2)^2))*(1 -
+       ((x[3] - 1/2)^2 + (x[4] - 1/2)^2))))^(2*H))
+ }
R> hyper <- fieldsim(R, Elelevel = 1, Rlevel = 5)

```

Acknowledgments

Part of this work was supported by the Interuniversity Attraction Pole (IAP) research network in Statistics P5/24. The Brouste's research was completed within laboratory LJK in Grenoble.

References

- Ayache A, Léger S, Pontier M (2002). “Drap Brownien Fractionnaire.” *Potential Analysis*, **17**, 31–43.
- Ayache A, Lévy-Vehel J (2000). “The Multifractal Brownian Motion.” *Statistical Inference for Stochastic Process*, **1**, 7–18.
- Bardet J, Lang G, Oppenheim G, Philippe A, Taqqu M (2003). “Generators of the Long-range Dependence Processes: A Survey.” In M Taqqu, G Oppenheim, P Doukhan (eds.), “Long range dependence,” pp. 579–624. Birkhäuser.
- Bégyn A (2006). *Generalized Quadratic Variations of Gaussian Processes: Limit Theorems and Applications to Fractional Processes*. Ph.D. thesis, Université de Toulouse.

- Benassi A, Bertrand P, Cohen S, Istas J (2000). “Identification of the Hurst Index of a Step Fractional Brownian Motion.” *Statistical Inference for Stochastic Process*, **3**, 101–111.
- Benassi A, Cohen S, Istas J (1998). “Identifying the Multifractal Function of a Gaussian Process.” *Statistics and Probability Letters*, **39**, 337–345.
- Benassi A, Jaffard S, Roux D (1997). “Elliptic Gaussian Random Processes.” *Revista Mathematica Iberoamericana*, **18**, 19–89.
- Biermé H (2005). *Champs Aléatoires : Autosimilarité, Anisotropie et Etude Directionnelle*. Ph.D. thesis, Université d’Orléans.
- Bonami A, Estrade A (2004). “Anisotropic Analysis of Gaussian models.” *Journal of Fourier Analysis and Applications*, **9**, 215–236.
- Coeurjolly J (2000). “Simulation and Identification of the Fractional Brownian Motion: A Bibliographical and Comparative Study.” *Journal of Statistical Software*, **5**(7), 1–53. URL <http://www.jstatsoft.org/v05/i07>.
- Dégérine S, Lambert-Lacroix S (2003). “Partial Autocorrelation Function of a Nonstationary Time Series.” *Journal of Multivariate Analysis*, **87**(1), 46–59.
- Fournier A, Fussel D, Carpenter L (1982). “Computer Rendering of Stochastic Models.” *Communication of the ACM*, **25**, 371–384.
- Frisch U, Parisi G (1985). “Turbulence and Predictability in Geophysical Fluid Dynamics and Climate Dynamics.” In M Ghil, R Benzi, G Parisi (eds.), “Proceedings of the International School of Physics E. Fermi,” pp. 84–88. North-Holland.
- Helgason S (1962). *Differential Geometry and Symmetric Spaces*. Academic Press.
- Herbin E (2006). “From N-parameter Fractional Brownian Motions to N-parameter Multifractal Brownian Motion.” *Rocky Mountain Journal of Mathematics*, **3**, 1249–1284.
- Houdré C, Villa J (2003). “An Example of Infinite Dimensional Quasi-helix.” *Contemporary Mathematics, AMS*, **336**, 195–201.
- Istas J (2005). “Spherical and Hyperbolic Fractional Brownian Motion.” *Electronic Journal of Probability*, **10**, 254–262.
- Istas J, Lang G (1997). “Quadratic Variations and Estimation of the Local Hölder Index of a Gaussian Process.” *Annales Institut Henri Poincaré, Probabilités et Statistiques*, **33**, 407–436.
- Kamont A (1996). “On the Fractional Anisotropic Wiener Fields.” *Journal of Probability and Mathematical Statistics*, **18**, 85–98.
- Kolmogorov A (1940). “Wiener’s Spirals and Some Other Interesting Curves in Hilbert’s Space. (German).” *C. R. (Doklady) Academy of Sciences URSS*, **26**, 115–118.
- Lacaux C (2004). “Real Harmonizable Multifractal Lévy Motions.” *Annales Institut Henri Poincaré, Probabilités et Statistiques*, **40**, 259–277.

- Leland W, Taqqu M, Willinger W, Wilson V (1994). “On the Self-Similar Nature of Ethernet Traffic (Extended Version).” *IEEE/ACM Transactions on Networking*, **2**(1), 1–15.
- Lévy P (1965). *Processus Stochastiques et Mouvement Brownien*. Gauthier-Villars.
- Mandelbrot B (1975). “Stochastic Models for the Earth’s Relief, the Shape and the Fractal Dimension of the Coastlines, and the Number-Area Rule for Islands.” In “Proceedings of the National Academy of Sciences,” 10, pp. 3825–3828.
- Mandelbrot B, Ness JV (1968). “Fractional Brownian Motions, Fractional Noises and Application.” *SIAM Review*, **10**, 422–437.
- Peitgen H, Saupe D (1988). *The Science of Fractal Images*. Springer-Verlag.
- Peltier R, Levy-Véhel J (1996). “Multifractal Brownian Motion: Definition and Preliminary Results.” *Technical Report RR 2645*, INRIA. URL <http://hal.inria.fr/inria-00074045/fr/>.
- Pentland A (1984). “Fractal-based Description of Natural Scenes.” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **6**, 661–674.
- Samorodnitsky G, Taqqu M (1994). *Stable non-Gaussian Random Processes: Stochastic Models with Infinite Variance*. Chapman & Hall, New York.
- Voss R (1985). “Random Fractal Forgeries.” *NATO ASI Series*, **F17**, 805–835.

Affiliation:

Alexandre Brouste

Laboratoire de Statistique et Processus

Avenue Olivier Messiaen

72000 Le Mans cedex 9, France

E-mail: Alexandre.Brouste@univ-lemans.fr

URL: http://www.univ-lemans.fr/sciences/statist/pages_persos/Brouste/

Jacques Istas

LJK, BP 53

38041 Grenoble cedex 9, France

E-mail: Jacques.Istas@imag.fr

URL: <http://ljk.imag.fr/membres/Jacques.Istas>

Sophie Lambert-Lacroix

LJK, BP 53

38041 Grenoble cedex 9, France

E-mail: Sophie.Lambert@imag.fr

URL: <http://ljk.imag.fr/membres/Sophie.Lambert>

Journal of Statistical Software

published by the American Statistical Association

Volume VV, Issue II

MMMMMM YYYY

<http://www.jstatsoft.org/>

<http://www.amstat.org/>

Submitted: yyyy-mm-dd

Accepted: yyyy-mm-dd