On simulation of manifold indexed fractional Gaussian fields

Alexandre Brouste,
LSP, Université du Maine,
Avenue Olivier Messiaen,
72000 Le Mans cedex 9, France.
Jacques Istas,
LJK - Université de Grenoble,
BP 53, 38041 Grenoble cedex 9, France.
Sophie Lambert-Lacroix,
LJK - Université de Grenoble,
BP 53, 38041 Grenoble cedex 9, France.

Abstract. To simulate fractional Brownian motion indexed by a manifold poses serious numerical problems: storage, computing time and choice of an appropriate grid. We propose an effective and fast method, valid not only for fractional Brownian fields indexed by a manifold, but for any Gaussian fields indexed by a manifold. The performance of our method is illustrated with different manifolds (sphere, hyperboloid).

Keywords. Manifold indexed fractional Brownian field, simulation, Hölder index.

Availability. The software FieldSim is an R package developed in R and C and that implements the procedures on which this paper focuses. It is available at http://ljk.imag.fr/membres/Sophie.Lambert

1 Introduction

Rough phenomena arise in various fields (e.g. [2, 4, 5, 6, 7]): texture simulations and image processing, natural scenes (clouds, mountains) simulations, fluid mechanics, financial mathematics, ethernet traffic ... Some phenomena, like ethernet traffic or financial data, are time-indexed. Other phenomena, like image processing, should be indexed by subsets of the Euclidean spaces $\mathbb{R}^2$ or $\mathbb{R}^3$. But, some other rough phenomena are not indexed by an Euclidean space, but by
a manifold. Let us mention for instance the cosmic microwave background ([?] or data on
http://www.fisica.uniroma2.it/cosmo/), or solar data [? ], that lead to spherical data. Real-world
textures are usually not supported by an Euclidean space, but by a surface. Applying an Euclidean
indexed texture on, say, a sphere, always generates artefacts. Generating manifold indexed textures
is therefore a challenge.

Fractional Gaussian fields, like fractional Brownian fields, are good candidates for modeling rough
phenomena. Fractional Gaussian fields may be indexed by an Euclidean space or a manifold. The
simulation of Euclidean indexed Gaussian fields poses a lot of problem, in particular storage and
complexity. Recently, [1] proposes a new and fast algorithm for simulating Euclidean indexed
Gaussian fields. The aim of this paper is to extend this algorithm to the manifold indexed case.
Compared to the Euclidean case, two problems emerge. First, there is no more a natural grid,
with equidistributed points. One therefore has to chose a grid and to adapt the algorithm to this
non-equidistributed grid. Second, the visualisation software (here R) imposes a visualisation grid
for representing the manifold. That means that the algorithm must adapt to this grid.

The simulation of a Gaussian random field reduces indeed to the simulation of a Gaussian random
vector on a given grid. On a given bounded manifold, there usually does not exist an equidistributed
grid, but there exists a uniform probability measure. Our algorithm contains three steps. We generate
a set of random points following a uniform density. This set is separated into two subsets. On the
first one, one generates a Gaussian random vector via an exact method. On the second subset,
one keeps only the neighbors to simulate a random Gaussian vector. At the end, the points of the
visualisation grid are considered as neighbors of previous points and a last simulation is done.

The paper proceeds as follows. In the second section, we present our procedure mifieldsim. In
the third section, we present our method on spheres and hyperboloid. Technical tools are postponed
at the end of the paper.

2 Method

After introducing some notations, we recall the both (accurate and refined) steps of the procedure
fieldsim proposed in [1]. Then we present the extension of this method to the processes indexed
by a manifold.

2.1 Notations and definitions

Let \((\mathcal{M}, g)\) be a \(C^\infty\)-complete Riemannian manifold of dimension 2. Let \(d_M\) be its geodesic distance. Let \(X(\cdot) = \{X(M), M \in \mathcal{M}\}\) be a real valued centered Gaussian field indexed by the manifold \(\mathcal{M}\). When \((\mathcal{M}, g)\) is the usual Euclidean space, one speaks of \(X\) as an Euclidean random field.

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

\[
R(M, M') = \langle X(M), X(M') \rangle = Cov \{X(M), X(M')\} \quad M, M' \in \mathcal{M}.
\]

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

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

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

We give here a summary of the procedure \texttt{fieldsim} that allows to simulate an Euclidean random field. In this case \(\mathcal{M}\) is a subspace like \([0, 1]^2\) for instance with the usual Euclidean norm. This procedure yields discretization of sample path of the Gaussian field over a (regular) space discretization of \(\mathcal{M}\), associated with any n.n.d. function \(R(\cdot, \cdot)\). It is consisted of the two following steps.

\textit{Accurate simulation step.} Given a (regular) space discretization \(\{M^i, i \in I\}\) of \(\mathcal{M}\), a sample of a centered Gaussian vector: \((X(M^i))_{i \in I}\) of covariance matrix \(R\) given by \(R_{i,j} = R(M^i, M^j)\), \(i, j \in I\), is simulated. This simulation is obtained by an algorithm based on Cholesky decomposition of the matrix \(R\) in the sequential manner.
Refined simulation step. Let $M$ be a new point at which we want to simulate the field. To simulate $X(M)$, one use only a set of neighbors $X_{N_M} = \mathcal{P}\{X(M^i), i \in N_M\}$, where $N_M$ is a neighbors indexes set of $M$, instead of all the simulated components (as in the accurate simulation step). Let $X_{X_{N_M}}(M)$ be the best linear combination of variables of $X_{N_M}$ approximating $X(M)$ in the sense that the variance of the innovation $\varepsilon_{X_{N_M}}(M) = X(M) - X_{X_{N_M}}(M)$ is minimum. The new variable $X(M)$ is obtained by $X_{X_{N_M}}(M) + \sqrt{\text{Var}(\varepsilon_{X_{N_M}}(M))}U$, where $U$ is a centered and reduced Gaussian variable independent of the already simulated components.

Note that the variable $X_{X_{N_M}}(M)$ and the variance $\text{Var}(\varepsilon_{X_{N_M}}(M))$ are completely determined by the covariance structure of the sequence $X(M), X(M^i), i \in N_M$. For storage and computing time, the accurate simulation step must concern only a small variables number whereas the second step can relate a larger variables number. That leads to an effective and fast method to simulate any Gaussian field.

In [1], the procedure is implemented in the software FieldSim (in R). A natural discretization space is of the form $(k2^{-J}, l2^{-J})$, $k, l = 0, \ldots, 2^J$ for $J$ a positive integer. The accurate step is applied for points $(k2^{-J_a}, l2^{-J_a})$, where $J_a$ is a level to be chosen (in general $J_a = 1$ or 2). The refined step is applied for the remainder.

2.3 The procedure mifieldsim

The previous procedure can be adapted to the case of fields indexed by a manifold. The main problem stands in the discretization grid choice. For instance, this choice is delicate for the sphere and can be related to the software that one wishes to use to represent the manifold. Moreover it is difficult to define a concept of finer grid such as in the case of field indexed by $[0, 1]^2$. It is why we adopt here a modified approach.

We denote by $S_g$ the set of the point of $\mathcal{M}$ at which we want to generate the process. This set choice can be induced by the software that one wishes to use to represent the manifold. Let us recall that we want to generate a path of field indexed by the manifold $\mathcal{M}$ (discretized at $S_g$) and with covariance function $R(\cdot, \cdot)$. To do that, we propose the following procedure.

The procedure mifieldsim
1. Let us generate a set $S_e$ of $N_e$ uniform random points on the manifold $\mathcal{M}$. As in the accurate simulation step of $\text{fieldsim}$, a sample of a centered Gaussian vector: $(X(M))_{M \in S_e}$ of covariance matrix associated with $R(\cdot, \cdot)$ is simulated.

2. Let us generate a set $S_r$ of $N_r$ uniform random points on $\mathcal{M}$. As in the refined simulation step of $\text{fieldsim}$, the new sample of a centered Gaussian vector: $(X(M))_{M \in S_r}$ is simulated using only $n$ neighbors (already generated).

3. The sample of a centered Gaussian vector: $(X(M))_{M \in S_g}$ is simulated using only $\text{nbNeighbor}$ neighbors (already generated).

The generation of uniform random points on the manifold as sphere or hyperboloid are given in Appendix 4.1. Let us remark that this procedure requires in addition to $R(\cdot, \cdot)$ the integers $N_e$, $N_r$ and $n$. A good choice for $N_e + N_r$ can be 4 times the cardinal of $S_r$. We need also to precise the concept of neighbors. A natural choice is to use the geodesic distance between two points. So the closest neighbors of some point $M$ are the points closest to $M$ according to this distance.

3 Numerical results

In this section, we illustrate the method proposed here through simulations for two manifolds: sphere and hyperboloid.

3.1 Some examples of Sphere indexed fractional fields

Let $S$ be the unit sphere of $\mathbb{R}^3$, and let $d_S$ be the geodesic distance. We give hereafter several examples of fields indexed by $S$.

The spherical fractional Brownian fields were introduced by [3] in the following way. There exists a centered Gaussian field called spherical fractional Brownian field (sfBf) whose covariance function is given by

$$R_{S_1}(M, M') = \frac{1}{2} \left\{ d_S^H(O, M) + d_S^H(O, M') - d_S^H(M, M') \right\},$$

where $O$ is any given point of $S$ if and only if $H \in (0, 1/2]$. 

5
We can show (see [?]) that the following functions
\[
R_{S^2}(M, M') = \exp \left( -d_{S}^2(M, M') \right),
\]
\[
R_{S^3}(M, M') = \ln \left( 1 + d_{S}^2(O, M) \right) + \ln \left( 1 + d_{S}^2(O, M') \right) - \ln \left( 1 + d_{S}^2(M, M') \right),
\]
\[
R_{S^4}(M, M') = \frac{1}{1 + d_{S}^2(M, M')},
\]
are covariance functions for \( H \in (0, 1/2] \).

Finally we can restrict the fractional Brownian fields indexed by \( \mathbb{R}^3 \) to the sphere. We obtain the following covariance function:
\[
R_{S^5}(M, M') = 1 - 2^{2H-1} \left( \sin \left( \frac{d_{S}(M, M')}{2} \right) \right)^{2H},
\]
where \( H \in (0, 1) \).

Figures 1 and 2 gives some representations of such fields. For each sphere, we have chosen \( N_e = 100 \) and \( N_r = 1000 \). The size of the discretized grid \( S_g \) is given by \( 6N_g^2 \) where \( N_g = 100 \).

### 3.2 Some examples of Hyperboloïd indexed fractional fields

Let \( \mathcal{H} \) be the hyperboloïd: \( \mathcal{H} = \{ x^2 + y^2 - z^2 = -1, \ z \geq 1 \} \) with geodesic distance \( d_{\mathcal{H}} \). Since \( \mathcal{H} \) is unbounded, we will simulate the field on a hyperbolic cap. We give hereafter several examples of fields indexed by \( \mathcal{H} \).

The hyperbolic fractional Brownian fields (hfBf) introduced by [3] have covariance function given by
\[
R_{H^1}(M, M') = \frac{1}{2} \left\{ d_{\mathcal{H}}^2(O, M) + d_{\mathcal{H}}^2(O, M') - d_{\mathcal{H}}^2(M, M') \right\},
\]
where \( O \) is any given point of \( \mathcal{H} \) if and only if \( H \in (0, 1/2] \).

We can show (see [?]) that the following functions
\[
R_{H^2}(M, M') = \exp \left( -d_{\mathcal{H}}^2(M, M') \right),
\]
\[
R_{H^3}(M, M') = \ln \left( 1 + d_{\mathcal{H}}^2(O, M) \right) + \ln \left( 1 + d_{\mathcal{H}}^2(O, M') \right) - \ln \left( 1 + d_{\mathcal{H}}^2(M, M') \right),
\]
\[
R_{H^4}(M, M') = \frac{1}{1 + d_{\mathcal{H}}^2(M, M')},
\]
are covariance functions for \( H \in (0, 1/2] \).

Figure 3 gives some representations of such fields. For each example, we have chosen \( N_e = 100 \) and \( N_r = 1000 \). The size of the discretized grid \( S_g \) is given by \( N_g^2 \) where \( N_g = 150 \).
Figure 1: On the left, Sphere indexed fractional fields with covariance function $R_{S_1}$ and Hölder index $H = 0.45$ (top), $H = 0.3$ (middle) and $H = 0.15$ (bottom). On the right, Sphere indexed fractional fields of Hölder index $H = 0.45$ with covariance function $R_{S_2}$ (top), $R_{S_3}$ (middle) and $R_{S_4}$ (bottom). All simulations are done with $N_c = 100$, $N_r = 1000$ and $N_g = 100$. 
4 Appendix

4.1 Uniform random generator on manifold

4.1.1 Spherical uniform random generator

We parameterize the unit sphere by the polar coordinate system

\begin{align*}
    x &= \cos \theta \sin \phi, \quad y = \sin \theta \sin \phi, \quad z = \cos \phi,
\end{align*}

where \( \phi \in [0, \pi] \) and \( \theta \in [0, 2\pi] \). To generate uniform random points on the unit sphere, we can use the following way. Firstly we generate \( \theta \) with an uniform distribution on \([0, 2\pi]\). Secondly (and independently of the first sample), we generate \( \phi \) as a random variable on \([0, \pi]\) with density \( \sin(\phi)/2 \).

4.1.2 Hyperboloïd uniform random generator

We parameterize the hyperboloïd by the hyperbolic coordinate system

\begin{align*}
    x &= \cos \theta \sinh \phi, \quad y = \sin \theta \sinh \phi, \quad z = \cosh \phi,
\end{align*}

where \( \phi \geq 0 \) and \( \theta \in [0, 2\pi] \). Let us remark that there does not exist uniform random generator on hyperboloïd since this space is not bounded. So one considers the hyperbolic cap defined by \([x^2 + y^2 - z^2 = -1, 1 \leq z \leq Z]\), where \( Z \) is some real in \((1, \infty)\). To generate uniform random points on the hyperbolic cap, we can use the following way. Firstly we generate \( \theta \) with an uniform distribution on \([0, 2\pi]\). Secondly (and independently of the first sample), we generate \( \phi \) as a random variable on \([0, \text{arccosh} Z]\) with density \( \sinh(\phi/(Z - 1)) \).

4.2 Using FieldSim

4.2.1 Implementation of FieldSim

The new version of FieldSim is a set of \( \mathbb{R} \) functions that allows performing simulations of manifold indexed fractional Gaussian fields with known covariance function. For the Euclidean random field,
see [1]. For the spherical or hyperboloid indexed fractional Gaussian fields, three classes of functions are implemented:

- Simulation functions `spheresim`, `spheresimgrid`, `hypersim` and `hypersimgrid` that performs simulations of the path of the manifold indexed Gaussian fields.

- Print functions `printsphere` and `printhyper` that give graphical representation of path of the manifold indexed Gaussian fields.

- C subroutine `vf3` performs the tasks that are consuming because of the number of loops.

The R environment is the only user interface. Simulation procedures calls the C subroutine `vf3` whose result is returned to R.

### 4.2.2 Simulations

In order to make it easier for the reader not used to R language, we detail the call to functions and the command used to produce graphical outputs. In the following, the prompt symbol is `R>`

**Sphere indexed fractional fields.** To simulate sphere indexed fractional fields, one needs to specify the Hurst parameter $0 < H < 1/2$ and the covariance function $R(M, M')$. For instance,

```r
R> set.seed(200)
R> d <- function(x){
R>   if (u<(-1))
R>     u <- -1
R>   if (u>1)
R>     u <- 1
R>   acos(u)}
R> R1 <- function(x){
R>   H <- 0.45
R>   1/2*(d(c(1,0,0,x[1:3]))^{2*H}+d(c(1,0,0,x[4:6]))^{2*H}-d(x)^{2*H})
```

9
Next

R> res1 <- spheresim(R1,Ne=100,Nr=100,Ng=10,nbNeighbor=4)

simulates the corresponding fields. The value $Ng$ gives the size of the grid for the graphical representation as $6N_g^2$. The result of the function `spheresim` is an R Object of class `list`. It contains the following elements:

- vectors $X$, $Y$ and $Z$ of $x$, $y$ and $z$ coordinates, and the vector $W$ of the simulated path of the process;
- matrices $W1,\ldots,W6$ of size $Ng \times Ng$ that give values of the simulated spherical field at the points of the visual grid;
- real time that gives the CPU time.

To give a graphical representation of the field, we can use

R> library(rgl)
R> library(RColorBrewer)
R> printsphere(res1)

Note that the procedure `printsphere` use needs the library `rgl` and `RColorBrewer` and works only with object result of `spheresim` procedure.

Next we simulate the field and compute the quadratic variations when the Hurst index is less than $3/4$. Note that in order to simulate the field on a specified grid one must use `spheresimgrid`.

R> Wg <- spheresimgrid(R1,Ne=100,Nr=100,nbNeighbor=4,S)$Wg
R> Wn <- t(matrix(Wg,nrow=(n-1),ncol=(n-1),byrow=TRUE))
R> Vn <- sum((Wn[-,-1]-Wn[-,-ncol(Wn)])^2)
R> x <- seq(1,nrow(Wn),2)
R> y <- seq(1,ncol(Wn),2)
R> Wn2 <- Wn[x,y]
R> Vn2 <- sum((Wn2[-,-1]-Wn2[-,-ncol(Wn2)])^2)
R> hatH <- 1 + log2(Vn2/Vn)/2
Hyperboloïd indexed fractional fields. To simulate hyperboloïd indexed fractional fields, one needs also to specify the Hurst parameter $0 < H \leq 1/2$ and the covariance function $R(M, M')$. For instance,

```r
R> set.seed(200)
R> d <- function(x){
R>   if (u<1){u<-1}
R>   acosh(u)}
R> R3 <- function(x){
R>   H <- 0.3
R>   1/2*(d(c(1,0,0,x[1:3]))^{2*H}+d(c(1,0,0,x[4:6]))^{2*H}-d(x)^{2*H})
R>   res2 <- hypersim(R3,Ne=100,Nr=100,Ng=10,nbNeighbor=4)
R>   printhyper(resh2)
```

Acknowledgements

Part of this work was supported by the Interuniversity Attraction Pole (IAP) research network in Statistics P5/24 and ANR-GDSA project.

References


Figure 2: Sphere indexed fractional fields with covariance function $R_{S^5}$ and Hölder index $H = 0.2$ (top left), $H = 0.45$ (top right), $H = 0.5$ (bottom left) and $H = 0.8$ (bottom right). All simulations are done with $N_e = 100$, $N_r = 1000$ and $N_g = 100$. 
Figure 3: On the left, Hyperboloid indexed fractional fields with covariance function $R_{H1}$ and Hölder index $H = 0.45$ (top), $H = 0.3$ (middle) and $H = 0.15$ (bottom). On the right, Hyperboloid indexed fractional fields of Hölder index $H = 0.45$ with covariance function $R_{H2}$ (top), $R_{H3}$ (middle) and $R_{H4}$ (bottom). All simulations are done with $N_e = 100$, $N_r = 1000$ and $N_g = 150$. 