

# Efficient parallel random field generator for large 3-D geophysical problems

Ludovic Räss<sup>a,b,c</sup>, Dmitriy Kolyukhin<sup>d</sup>, Alexander Minakov<sup>e</sup>

<sup>a</sup>*Stanford University, Geophysics Department, 397 Panama Mall, Stanford CA 94305, USA.*

<sup>b</sup>*Institute of Earth Sciences, University of Lausanne, Lausanne, Switzerland.*

<sup>c</sup>*Swiss Geocomputing Centre, University of Lausanne, Lausanne, Switzerland*

<sup>d</sup>*Institute of Petroleum Geology and Geophysics SB RAS, pr. Koptiyuga, 3, 630090, Novosibirsk, Russia.*

<sup>e</sup>*Centre for Earth Evolution and Dynamics (CEED), University of Oslo, Norway.*

---

## Abstract

We present an efficient implementation of the method for sampling spatial realisations of a 3-D random fields with given power spectrum. The method allows for a multi-scale resolution and approaches well for parallel implementations, overcoming the physical limitation of computer memory when dealing with large 3-D problems. We implement the random field generator to execute on graphical processing units (GPU) using the CUDA C programming language. We compare the memory footprint and the wall-time of our implementation to FFT-based solutions. We illustrate the efficiency of the proposed numerical method using examples of an acoustic scattering problem which can be encountered both in controlled-source and earthquake seismology. In particular, we apply our method to study the scattering of seismic waves in 3-D anisotropic random media with a particular focus on P-wave coda observations and seismic monitoring of hydrocarbon reservoirs.

*Keywords:* Geophysics; Geostatistics; Seismology; Computational Methods; Parallel and High-Performance Computing.

---

## Highlights

- 3-D Gaussian random fields generation

---

*Email address:* ludovic.rass@gmail.com (Ludovic Räss)

- GPU-based parallel algorithm provided
- Application to study the scattering of seismic P-wave coda in 3-D and monitoring of reservoirs

### **Computer Code Availability**

The presented GRFS (exponential and Gaussian covariance) algorithm and computer codes (serial Matlab & parallel GPU-based CUDA C) are provided as supplementary material and available for download from Bitbucket at <https://bitbucket.org/lraess/grfs> and from the Swiss Geocomputing Centre website <http://wp.unil.ch/geocomputing/software/grfs/>. Both the exponential and Gaussian covariance implementations are shared. The GPU cuFFT-based benchmarking algorithm as well as the GPU-based 3-D acoustic wave propagation software can be obtained upon request from the authors.

### **Author's Contribution**

All authors contributed to the overall realisation and the writing of this study. DK provided the statistical model used in the numerical simulation. LR developed the numerical algorithms, performed the GPU-based calculations and processed the results. AM suggested geophysical applications and the overall design of the paper.

### **Acknowledgments**

The authors thank Thomas Bodin as well as two anonymous reviewers for their valuable feedback that permitted to enhance the first version of this manuscript. LR acknowledges support from the Swiss National Science Foundation's Early Postdoc.Mobility Fellowship 178075 and computational resources from the Swiss Geocomputing Centre. DK acknowledges financial support from RFBR (grants no. 15-55-20004, 18-05-00031). AM acknowledges support by the Research Council of Norway through its Centers of Excellence funding scheme, project 223272.

## 1. Introduction

Incomplete data and complex nature of geophysical phenomena leads to the need for statistical modelling. The results of this approach can also be used, for example, for sensitivity analysis or uncertainty quantification. Stochastic simulations of random fields with multi-scale resolution have found application in studies of turbulent flows [1, 2, 3], flow in porous media [4], seismic volcanology [5, 6], large-scale density structure of the universe [7] and other geophysical and astrophysical problems. Some other geoscience applications and simulation techniques are described in the monograph by Christakos [8].

One of the important applications of the random field simulation in computational geophysics is linked to numerical mechanical modelling. Numerical analysis of large mechanical problems with severe nonlinearities such as flow localisation in two-phase deformable media [9] or simulation of fracture networks and dike swarms [10] rely on a model initialisation procedure including computation of random field of small parameter perturbation (e.g. porosity or rock cohesion). The important requirement is that both large and small spatial scales must be accurately resolved by the numerical grid to obtain a physically meaningful solution.

Some geophysical inverse problems directly target on statistical properties of the medium (e.g. ocean temperature fluctuation spectra, lithosphere gravity-topography spectral ratios, scattering strength and attenuation spectra from seismic coda waves). For this class of problems proving the parameter sensitivity and model resolution requires a random field simulator (RFS) reproducing observed statistical behaviour of the physical process such as correlation function or power spectrum within a broad range of scales. The accurate statistics is also required for inverse geophysical problems formulated as a Bayesian inference problem with physically-based prior distribution [11].

Cholesky decomposition of the covariance matrix is perhaps the most well-known method to generate the realisations of Gaussian random fields. This method is based on a clear idea, simple in use, and allows in addition to per-

form conditional simulations to adapt the measurement data. However, as the number of grid points increases, this method becomes very computationally time consuming. The turning-band method [12] and spectral method [13] are among other frequently utilised approaches for modelling random fields. An  
35 efficient method for sampling random realisations, based on fast Fourier transform (FFT) moving-average method was developed in Le Ravalec et al. [14]. The comparative analysis of Fourier-wavelet simulation method against spectral method with different randomisations of wavenumber space is performed in Kramer et al. [15]. The Karhunen-Loève expansion based on finding the  
40 eigenvalues and eigenvectors of the covariance function can be used to simulate non-homogeneous random fields [16]. In Phoon et al. [17] this technique was extended for simulation of strongly non-Gaussian random fields. Another method that allows one to model more complex non-homogeneous and non-Gaussian fields is to use Polynomial Chaos extension [18].

45 A number of methods for indicator random fields simulation that can have discrete values have been developed [19]. For simple geological models, the simulation of the indicator random fields with given mean values and the covariance function can be achieved using a truncated Gaussian simulation (TGS). This method is based on the use of an auxiliary Gaussian random field constructed  
50 by the methods described above. For more complex geological models, a more general truncated PluriGaussian simulation (TPGS) was developed [20]. TGS and TPGS are based on the reproduction of two-point statistics (covariance or variogram functions). These methods have a number of limitations when modelling structures with a complex curvilinear geometry (for example, media with  
55 channels). In recent years, the usages of statistical modelling methods relying on multipoint statistics increased [21, 22]. One more approach to statistical modelling is based on the use of optimisation methods such as the simulated annealing, which requires to consider some objective functions [23]. For example, Tran et al. [24] use this approach to simulate the spatial distribution of  
60 a fracture network. A new Gibbs sampler for simulation of Gaussian Markov random fields on large lattice was developed recently in Marcotte and Allard

[25].

Fast spectral methods based on the FFT are traditionally used in seismology [26, 5, 27]. The disadvantage of this approach is the need to generate the values  
65 of the realisation of a random field in all nodes of the computational domain. Moreover, a computation on an irregular grid is difficult using this method.

We present here a parallel implementation of the method based on spectral representation described in Sabelfeld [28]. Advantages of this method are the possibility of random field simulation on an arbitrary grid and the simplicity  
70 of parallel implementation of the algorithm. The method is flexible and is also applicable for arbitrary anisotropic spectrums. Moreover, at present this simulation technique is well studied. For example, ergodicity properties were studied by Kramer et al. [15].

Serial algorithms are bottleneck for three-dimensional (3-D) computations.  
75 Routines that rely on local operation only can perform very efficiently on actual parallel hardware, such as graphical processing units (GPUs). Calculating every grid point of the computational domain on a distinct hardware thread permits thus to achieve a fast time to solution using performance-oriented software. In particular, the method presented in this paper is shown to be more flexible for  
80 large simulations than FFT-based methods.

We provide a recipe to efficiently compute a 3-D random fields with a known power spectrum or correlation function making a particular focus on geophysical applications. The presented GRFS (exponential and Gaussian covariance) algorithm and computer codes (Matlab & GPU-based CUDA C) are  
85 provided as supplementary material and available for download from Bitbucket at <https://bitbucket.org/lraess/GRFS> and from the Swiss Geocomputing Centre website <http://wp.unil.ch/geocomputing/software/grfs/>. The GPU cuFFT-based benchmarking algorithm as well as the GPU-based 3-D acoustic wave propagation software can be obtained upon request from the authors.

90 **2. Random Field Generator Based on Spectral Representation**

In this work we assume that the scalar random field  $f(\mathbf{x})$  is statistically homogeneous with a mean value  $m_f(\mathbf{x})$ . The probability density function of a scalar Gaussian random field  $f(\mathbf{x})$  in the points  $\mathbf{x} = \{\mathbf{x}_1, \dots, \mathbf{x}_n\}$

$$p(f(\mathbf{x}_1), \dots, f(\mathbf{x}_n)) = \frac{1}{(2\pi \det C_{ff})^{1/2}} \exp \left\{ -\frac{1}{2} (f(\mathbf{x}) - \mathbf{m})^T C_{ff}^{-1} (f(\mathbf{x}) - \mathbf{m}) \right\} \quad (1)$$

is fully defined by its mean values  $\mathbf{m} = \{m_1, \dots, m_n\}$  and the covariance function  $C_{ff}(\mathbf{x}_i, \mathbf{x}_j) = \langle (f(\mathbf{x}_i) - m_i)(f(\mathbf{x}_j) - m_j) \rangle$ . We use bold font for vectors,  $\langle \cdot \rangle$  – is a mathematical expectation.

The covariance function and the spectrum of homogeneous random fields are related through the equation:

$$S_{ff}(\mathbf{k}) = \frac{1}{8\pi^3} \int_{R^3} e^{-i(\mathbf{r}, \mathbf{k})} C_{ff}(\mathbf{r}) d\mathbf{r} . \quad (2)$$

The simulation formula based on the spectral randomisation method has the form:

$$f(\mathbf{x}) = m_f(\mathbf{x}) + \frac{\sigma_f}{\sqrt{N_h}} \sum_{i=1}^{N_h} [\xi_i \cos(\mathbf{k}_i, \mathbf{x}) + \eta_i \sin(\mathbf{k}_i, \mathbf{x})] \quad (3)$$

and can be used for random field generation [28]. Here  $\xi_i$  and  $\eta_i$  – are mutually independent and independent on  $\mathbf{k}_i$  random numbers with zero mean and variance equal to unity,  $(\mathbf{k}_i, \mathbf{x})$  – is the scalar product of the vectors  $\mathbf{k}_i$  and  $\mathbf{x}$ . Following Sabelfeld [28], the wave vector  $\mathbf{k}$  is sampled according to the probability density

$$p(\mathbf{k}) \propto \frac{S_{ff}(\mathbf{k})}{\int_{R^3} S_{ff}(\mathbf{k}) d\mathbf{k}} . \quad (4)$$

The central limit theorem provides that  $f(\mathbf{x})$  will converge to Gaussian distributions as the number of harmonics  $N_h \rightarrow \infty$  [28].

The presented approach is general and can be applied for any homogeneous covariance  $C_{ff}$  and corresponding  $S_{ff}$ . The key insight in this technique is the sampling of the wave vector  $\mathbf{k}$  according to the probability density (4). The general approaches for the sampling are described for example in Rubinstein [29] and Sabelfeld [28].

In this work, we mainly consider an anisotropic exponential covariance which is most often used in geosciences applications

$$C_{ff}(\mathbf{r}) = \sigma_f^2 \exp \left\{ - \left( \frac{r_1^2}{I_1^2} + \frac{r_2^2}{I_2^2} + \frac{r_3^2}{I_3^2} \right)^{1/2} \right\}, \quad (5)$$

115 where  $\mathbf{r} = (r_1, r_2, r_3)$  – is a separation vector,  $\sigma_f$  – is a standard deviation and  $I_j$ ,  $j = 1, 2, 3$  – are correlation lengths in  $j$ -th direction. Following Eyink and Goldenfeld [30] the correlation length is defined as:

$$I_j = \frac{1}{C_{ff}(0)} \int_0^\infty C_{ff}(r_j) dr_j. \quad (6)$$

Note, that other definitions of correlation length also appear in the scientific literature [31, 32]; for example, the length  $r$  for which

$$C_{ff}(\mathbf{r}) = C_{ff}(0)/e. \quad (7)$$

120 The covariance function (5) corresponds to the spectrum:

$$S_{ff}(\mathbf{k}) = \frac{\sigma_f^2 I_1 I_2 I_3}{\pi^2 (1 + I_1^2 k_1^2 + I_2^2 k_2^2 + I_3^2 k_3^2)^2}, \quad (8)$$

where  $\mathbf{k} = (k_1, k_2, k_3)$  is a wave vector.

To generate the realisations of the vector  $\mathbf{k}$ , the following formulas are used:

$$\begin{cases} k_1 = k \cos(\theta)/I_1, \\ k_2 = k \cos(\phi) \sin(\theta)/I_2, \\ k_3 = k \sin(\phi) \sin(\theta)/I_3. \end{cases} \quad (9)$$

The scalar value  $k$  is distributed with the probability distribution

$$p(k) = \frac{4k^2}{\pi(1 + k^2)^2}. \quad (10)$$

Its realisations can be generated using the rejection method, described in Rubinstein [29]. The value of the angle  $\phi$  is sampled uniformly in the interval  
125  $[0, 2\pi]$ , the value  $\theta$  is distributed with the probability density

$$p(\theta) = \frac{1}{2} \sin(\theta), \quad \theta \in [0, \pi]. \quad (11)$$

The corresponding simulation formula has the form

$$\theta = \arccos(1 - 2\lambda) , \quad (12)$$

where  $\lambda$  – is a random value uniformly distributed in  $[0, 1]$ . The implementation of the anisotropic exponential covariance Gaussian random field is presented in

130 Algorithm 1 (Section 4).

For comparison we also consider the isotropic Gaussian correlation function

$$C_{ff}(\mathbf{r}) = \sigma_f^2 \exp \left\{ -\frac{r^2}{l_f^2} \right\} , \quad (13)$$

with correlation length  $I_f = l_f \frac{\sqrt{\pi}}{2}$ . The corresponding spectrum [33] has the form:

$$S_{ff}(\mathbf{k}) = \frac{\sigma^2 l_f^3}{8\pi^{3/2}} \exp \left( -\frac{k^2 l_f^2}{4} \right) . \quad (14)$$

In this case, the scalar wave number is proportional to:

$$p(k') = k'^2 \exp \left( -\frac{k'^2}{2} \right) , \quad k' = k \frac{l_f}{\sqrt{2}} . \quad (15)$$

135 These numbers can be generated by using the rejection method [29]. The implementation of the isotropic Gaussian covariance alternative is presented in Algorithm 2 (Section 4).

### 3. Random Field Generator Based on Fast Fourier Transform

Further, the random field modelling method presented in the previous section is compared with the most effective to our knowledge and frequently used  
140 method for modelling of homogeneous random fields based on the Fast Fourier transform [34, 14, 35].

Direct  $F = FFT(f)$  and inverse  $f = FFT^{-1}(F)$  Fast Fourier transformations are:

$$F(j) = \sum_{k=0}^{N-1} f(k) \exp(2\pi i k j / N) , \quad (16)$$

$$145 \quad f(k) = \sum_{j=0}^{N-1} F(j) \exp(-2\pi i k j / N) , \quad (17)$$



where  $N$  is the number of sampled points.

The simulation formula takes the following form:

$$f(\mathbf{x}) = m_f(\mathbf{x}) + FFT^{-1} \left[ \sqrt{FFT(C_{ff}(\mathbf{x}))FFT(Z(\mathbf{x}))} \right], \quad (18)$$

where  $Z(\mathbf{x})$  are independent Gaussian random numbers with zero mean and unit variance.

150 It is important to emphasise that due to the periodicity of the FFT within the simulation we can only use the  $f(\mathbf{x})$  values generated for the inner subdomain. In consequence, the realisations should be oversized at the beginning and at the end of the domain; the near-border cells must be removed [14]. We recommend using an indent distance equal to at least two correlation lengths.

#### 155 4. Numerical Implementation

The motivation driving the development of the random field generator based on spectral representation is to propose an algorithm that runs efficiently and scales on parallel hardware, such as GPUs. These many-core accelerators are capable to efficiently perform a large number of identical tasks in parallel. This is of particular interest when performing simulations on large computational domains, inherent to 3-D calculations. However, efficient parallel algorithms strongly rely on local operations. For optimal performance, each parallel process (also called thread) may perform the workflow tasks independently, implying no communication with neighbouring threads, neither waiting on other processes to accomplish their tasks first. The proposed method to generate Gaussian random fields based on the spectral representation fulfils both aforementioned requirements; only the spatial coordinates of each grid point within the computational domain are mandatory and can be determined in a straight forward way. The inherent parallelism of the approach is thus very well suited for GPU computing. Each thread is responsible to compute one grid point of the domain based on its coordinate, without need of any communication neither global reduction operations. Irregular grid spacing can easily be obtained by choosing indices of spatial coordinates to be non-uniform.

---

**Algorithm 1** Gaussian random field simulator – Exponential covariance

---

```
1:  $C = \text{sigf}/\text{sqrt}(N_h)$ 
2: for (ih = 0; ih <  $N_h$ ; ih++) do
3:   fi =  $2\pi \text{rand}$ 
4:   flag = 1
5:   while (flag = 1) do
6:     k =  $\tan(\pi \cdot 0.5 \text{rand})$ 
7:     d =  $k^2/(1 + k^2)$ 
8:     if (rand < d) then
9:       flag = 0
10:    end if
11:  end while
12:  theta =  $\text{acos}(1 - 2 \text{rand})$ 
13:   $V_1 = k \sin(\text{fi}) \sin(\text{theta})/I_1$ 
14:   $V_2 = k \cos(\text{fi}) \sin(\text{theta})/I_2$ 
15:   $V_3 = k \cos(\text{theta})/I_3$ 
16:  a = randn
17:  b = randn
18:  compute_kernel_1 <<< grid, block >>> (); cudaDeviceSynchronize()
19: end for
20: compute_kernel_2 <<< grid, block >>> (); cudaDeviceSynchronize()
```

---

---

**Algorithm 2** Gaussian random field simulator – Gaussian covariance

---

```
1:  $C = \text{sigf}/\text{sqrt}(N_h)$ 
2:  $lf = 2 I/\text{sqrt}(\pi)$ 
3: for ( $ih = 0; ih < N_h; ih ++$ ) do
4:    $fi = 2 \pi \text{rand}$ 
5:    $flag = 1$ 
6:   while ( $flag = 1$ ) do
7:      $k = k_{\max} \text{rand}$ 
8:      $d = k^2 \exp(-0.5 k^2)$ 
9:     if ( $2 \text{rand} \exp(-1) < d$ ) then
10:       $flag = 0;$ 
11:    end if
12:  end while
13:   $k = k \text{sqrt}(2)/lf$ 
14:   $theta = \text{acos}(1 - 2 \text{rand})$ 
15:   $V_1 = k \sin(fi) \sin(theta)$ 
16:   $V_2 = k \cos(fi) \sin(theta)$ 
17:   $V_3 = k \cos(theta)$ 
18:   $a = \text{randn}$ 
19:   $b = \text{randn}$ 
20:   $\text{compute\_kernel\_1} \lll \lll \text{grid, block} \ggg (); \text{cudaDeviceSynchronize}()$ 
21: end for
22:  $\text{compute\_kernel\_2} \lll \lll \text{grid, block} \ggg (); \text{cudaDeviceSynchronize}()$ 
```

---

---

**Algorithm 3** compute\_kernel\_1

---

```
1:  $\text{tmp} = dx((ix + 1) - 0.5)V_1 + dy((iy + 1) - 0.5)V_2 + dz((iz + 1) - 0.5)V_3$ 
2:  $Yf = Yf + a \sin(\text{tmp}) + b \cos(\text{tmp})$ 
```

---

---

**Algorithm 4** compute\_kernel\_2

---

```
1:  $Yf = C Yf$ 
```

---

The algorithm is composed of three main blocks; 1) the host code (executed  
 175 by the host – the central processing unit or CPU) and common to the entire spa-  
 tial domain (Algorithm 1 – exponential covariance or Algorithm 2 – Gaussian  
 covariance) containing the outer loop over a given (large) number of harmonics  
 $N_h$ , precomputing the wave vector, and calling the two GPU functions (also  
 called kernels) named `compute_kernel`; 2) the first GPU compute kernel (Al-  
 180 gorithm 3) calculates the random field values in parallel (3) for every thread  
 based on the wave vectors input; 3) the second compute kernel (Algorithm 4)  
 multiplies the random field values by a constant and must be located outside  
 the outer loop over the harmonics.

We implemented the algorithm using C and the CUDA extension in order  
 185 to execute the application in parallel on Nvidia GPUs. CUDA provides as main  
 feature a vectorised framework allowing to execute the serial loop over each  
 spatial dimension in parallel by the mean of their *thread*, *block* and *grid* layout  
 [36].

We generate uniformly distributed in  $[0,1]$  random numbers ('rand') within  
 190 the algorithm using the C built-in pseudo random number generator `rand()`  
 and normalise the output (using `RAND_MAX`) to the range  $[0,1]$  within the  
`RND()` macro. To obtain standard normal random numbers ('randn') in the  
 range  $[-\infty, \infty]$ , we rely on the Box-Muller algorithm [37] within the `RNDN()`  
 macro implementation:

$$\text{randn} = \sqrt{-2 \log(\lambda_1)} \cos(2\pi\lambda_2) \quad (19)$$

195 where  $\lambda_1$  and  $\lambda_2$  are two independent random numbers uniformly distributed  
 in  $[0,1]$  and obtained via the 'rand' routine.

Both single and double precision arithmetic is available to simulate the ran-  
 dom field within the routines. Some basic sensitivity analysis suggests that  
 single precision arithmetic delivers comparable results as double precision calcu-  
 200 lation for the here described purpose. This provides a non-negligible advantage  
 since single precision arithmetic calculations require twice less memory to be  
 transferred when compared to double precision. Single precision calculations

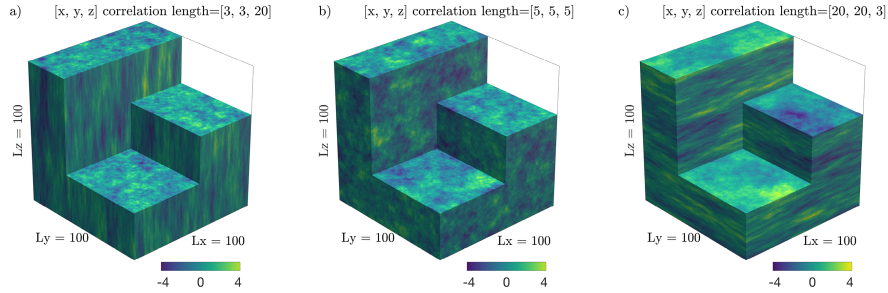


Figure 1: Simulated random field for different correlation lengths in all three spatial dimensions within a cube of size  $[100, 100, 100]$  in  $x$ ,  $y$  and  $z$  direction, respectively. a) Vertical tube-like random media, with correlation length of  $[3, 3, 20]$  in  $x$ ,  $y$  and  $z$  directions, respectively. b) Isotropic random media, with correlation length of  $[5, 5, 5]$  in all three directions. c) Horizontal planar-like random media, with correlation length of  $[20, 20, 3]$  in  $x$ ,  $y$  and  $z$  directions, respectively.

are thus twice more efficient in terms of execution time compared to double precision ones for memory bounded algorithms.

## 205 5. Results

### 5.1. Anisotropic Random Field Realisations

The outcome of the random field simulator are 3-D domains that depict various correlation lengths in all three spatial dimensions (Figure 1). The user defined input variables are the box size, the standard deviation of the Gaussian random field, the  $x$ ,  $y$  and  $z$  correlation lengths and the number of harmonics  
210  $(N_h)$ .

### 5.2. Benchmarks

Within this study we mainly showcase the usage of the exponential covariance function to generate Gaussian random fields, However, the approach we  
215 propose here generalises to other covariance functions, such as the Gaussian isotropic covariance (Figure 2). We perform three different benchmarks to validate both the presented approach for simulating Gaussian random fields using

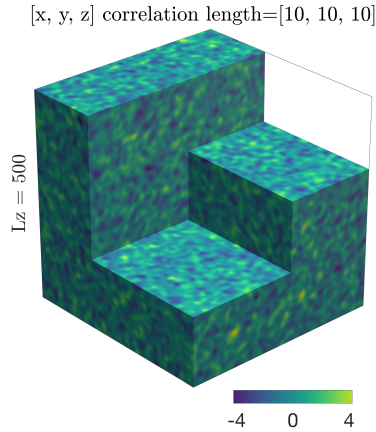


Figure 2: Simulated random field using an isotropic Gaussian covariance function with correlation length of 10 in all 3 spatial dimensions within a cube of size [500, 500, 500] in  $x$ ,  $y$  and  $z$  direction, respectively.

a) an anisotropic exponential covariance function and b) an isotropic Gaussian covariance function and thus highlight the potential of the method. Finally, we discuss the numerical implementation on GPUs and report performance results comparing our random field generator based on spectral representations to well-established FFT-based routines in terms of wall-time and overall memory footprint.

We firstly compute the covariance function of the random field realisations based on an anisotropic exponential covariance using spatial averaging (Figure 3a, d, g, j, m, ‘Simulated’) and compare it to the estimated covariance function using the analytical expression (5) for various number of harmonics ( $N_h$ ) (Figure 3a, d, g, j, m, ‘Analytical’). The accuracy of the simulated spatial realisation depends on the number of harmonics in the summation (3). We sample spatial realisations of a 3-D isotropic random field on a regular grid with constant grid step (correlation length = [5, 5, 5] – as in Figure 1b), to simplify the estimation of covariance function. In this case, the covariance can be computed over realisations just along one spatial direction. For correct spatial averaging the size of the domain must be significantly larger than the correlation lengths. Moreover,

235 in order to achieve an accurate comparison to the analytical estimation several  
 correlation lengths have to be covered by the linear sizes of the computational  
 and the estimated domains. We test 5 different number of harmonics ( $N_h$ )  
 ranging from  $10^2$  (Figure 3a–c) to  $10^5$  (Figure 3m–o). We report for every  $N_h$   
 realisation the corresponding spatial distribution for both the  $xy$  (Figure 3b, e,  
 240 h, k, n) and the  $xz$  (Figure 3c, f, i, l, o) slice. This experiment clearly shows the  
 convergence of the method when a sufficiently large number of harmonics ( $N_h$ )  
 is performed; the periodic noise pattern (aliasing) consequently vanishes from  
 the spatial random distribution and the covariance estimations  $C(r)$  plotted as  
 function of the spatial radius  $r$  are in good agreement with the analytical results.

245 We secondly evaluate the sensitivity of the simulated random fields based on  
 (a) an anisotropic exponential covariance function to the numerical grid reso-  
 lution (Figure 4). We consider 5 different realisations for numerical resolutions  
 ranging from  $32^3$  (Figure 4a–c) up to  $1024^3$  (Figure 4m–o) grid points in 3-D.  
 For each realisation we report the covariance function to be compared to the  
 250 analytical estimation (Figure 4a, d, g, j, m) as well as the spatial random field  
 distribution for both the  $xy$  (Figure 4b, e, h, k, n) and the  $xz$  (Figure 4c, f, i, l,  
 o) slices. We observe no significant influence of the numerical spatial resolution  
 on the results affecting the ‘Analytical’ and ‘Simulated’ covariance functions.  
 However, the spatial distribution of the random field is impacted by the low  
 255 resolution realisations; variations below the grid resolution ( $dx$ ) will not be de-  
 tected. These tests indicate that our method allows to reproduce accurately the  
 statistics of the random field even with a sparse grid resolution.

We in addition stress that the reported simulated random field covariance  
 functions are not equal to 1 for  $r = 0$  ( $C(0) \neq 1$ ) in case a single and thus  
 260 identical random number ( $\lambda_1 = \lambda_2$ ) is used within the Box-Muller algorithm  
 (19).

We repeat this second benchmark for the random field realisations based on  
 (b) an isotropic Gaussian covariance (Figure 5), and only show one realisation for  
 a numerical resolution of  $512^3$  grid points in 3-D. We benchmark the covariance  
 265 function to be compared to the analytical estimation (Figure 5a) as well as the

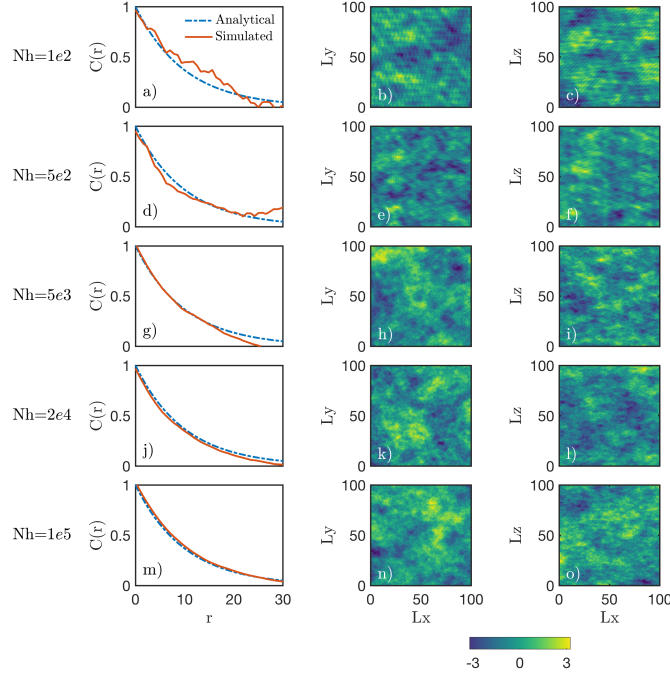


Figure 3: Comparison of covariance functions  $C(r)$  as function of spatial radius  $r$  for the simulated random fields versus the analytical expression. We compare the influence of the number of harmonics ( $N_h$ ) on the random field realisations for the isotropic model configuration (Figure 1b). We utilise a numerical grid resolution of  $512^3$  grid points in 3-D and constant grid steps.

spatial random field distribution for both the  $xy$  (Figure 5b) and the  $xz$  (Figure 5c) slices. We report the accuracy of the random field generator based on spectral representations to simulate Gaussian covariance function and confirm the ability of the approach to handle various covariance functions.

270 We compare the performance of our random field generator relying on spectral representations with an effective and frequently used method based on the Fast Fourier Transform (see Section 3). To this end, we implemented a 3-D FFT-based random field generator using the CUDA FFT library cuFFT [36]. Prior to evaluate the performance between the two approaches (spectral real-



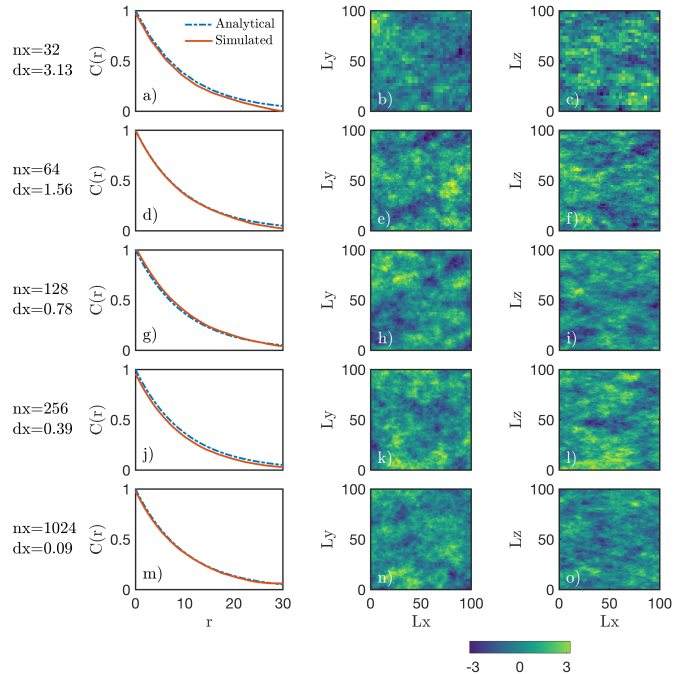


Figure 4: Comparison of covariance functions  $C(r)$  as function of spatial radius  $r$  for the simulated random fields versus the analytical expression. We compare the influence of refining the numerical grid resolution in 3-D on the simulated random field for the isotropic model configuration (Figure 1b).

275 isations and FFT), we benchmark the FFT-simulated exponential covariance  
function versus the analytical expression (Figure 6). We show that our FFT-  
based implementation is in good agreement with the analytical expression in  
this example involving  $512^3$  grid points. Note that we had to increase the do-  
main size to include 50 correlation lengths in order to converge towards accurate  
280 results.

We assess the performance of the random field simulator based on spec-  
tral representations (Section 2) comparing the memory footprint (Figure 7) as  
well as the wall-time (Figure 8) to the FFT-based simulator (Section 3). We  
report that both in single (SP) and double precision (DP) arithmetic the ran-

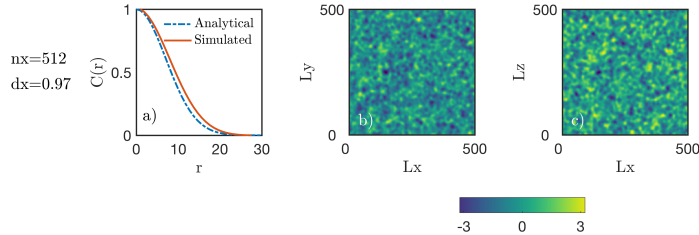


Figure 5: Comparison of covariance functions  $C(r)$  as function of spatial radius  $r$  for the simulated random fields versus the analytical expression. We compare the random field realisations for the model implementing the isotropic Gaussian covariance function (Figure 2). We utilise a numerical grid resolution of  $512^3$  grid points in 3-D and constant grid steps.

dom field generator based on spectral representation (SRep) consistently utilises  
 285 about seven times less memory compared to the FFT-based approach (Figure 7).  
 Also, both methods scale linearly with their respective memory footprints. The  
 discrepancy in the memory footprint between the two methods mainly comes  
 from the fact that only one 3-D field is mandatory for the spectral represen-  
 290 tation approach (20), while the FFT-based approach requires a more complex  
 data layout involving complex numbers as well (21). We evaluated the memory  
 footprint (in GB) for the spectral representation  $m_{\text{SRep}}$  and for the FFT-based  
 $m_{\text{FFT}}$  approaches, respectively, as following:

$$m_{\text{SRep}} = \frac{n_x n_y n_z n_p}{1024^3}, \quad (20)$$

$$m_{\text{FFT}} = \frac{(5n_x n_y n_z + 4n_x n_y (n_z/2 + 1)) n_p}{1024^3}, \quad (21)$$

295 where  $n_x, n_y, n_z$  is the grid resolution in the 3 spatial directions,  $x, y, z$ , respec-  
 tively, and  $n_p$  is the arithmetic precision in bytes (single=4, double=8).

Benefits of lower memory utilisation are especially noticeable for large prob-  
 lem size as the memory footprint determines the maximal problem size possible  
 to resolve on a single GPU. The approach relying on spectral representation  
 300 permits to resolve a 3-D problem of  $1792^3$  versus only  $1024^3$  for the FFT-based

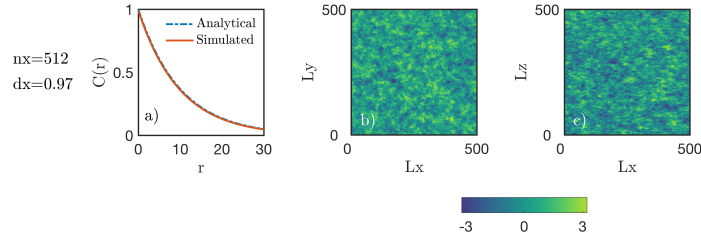


Figure 6: Comparison of covariance functions  $C(r)$  as function of spatial radius  $r$  for the FFT-based simulated random fields versus the analytical expression. We compare the random field realisations for the model implementing the anisotropic exponential covariance function. We utilise a numerical grid resolution of  $512^3$  grid points in 3-D and constant grid steps and a correlation length of  $[10,8,5]$  in  $x$ ,  $y$  and  $z$  direction, respectively.

approach – close to a factor 2 (Figure 7).

We then utilise the wall-time metric to report the time it takes in seconds to compute one realisation of the Gaussian random field using both approaches (Figure 8). The timings are in good agreement between the two approaches, although the spectral representation in double precision is slightly off compared to the three other cases. A general trend shows the latency bound effects of all 4 tested configuration up to a problem size of about  $128^3$  in 3-D on the GPU. Then, we report a linear increase in time to solution for all methods and considered arithmetic precisions. We employed  $N_h = 10'000$  for the spectral representation approach for this benchmark. Faster time to solution could be obtained by decreasing  $N_h$ , but this may lead to increased aliasing issues (see Section 5.2). Note that the scaling does not break even for the highest targeted resolution beyond the maximal domain size the FFT-based approach can handle.

We finally report the efficiency of the random field generator implementation on GPUs using using the effective memory throughput as metric [38]. The effective memory throughput  $MTP_{\text{eff}}$  evaluates the optimality of non-redundant data transfers from and to the GPU memory. The entire algorithm (Algorithm

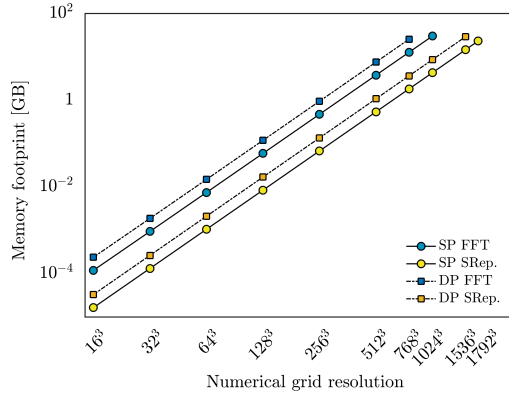


Figure 7: Total memory footprint [GB] of the spectral representation (SRep) and FFT-based (FFT) 3-D random field GPU simulator. We report the global memory utilisation, varying the problem size from  $16^3$  to  $1792^3$  for both single precision (SP) and double precision (DP) arithmetic. The spectral representation approach utilises only one 3-D field (20), while the FFT-based approach utilises the equivalent of 5 3-D fields (21).

1) is composed of 3 main blocks reported in Section 4. Lines 3–17 in Algorithm 1 generate 5 scalars ( $V_1, V_2, V_3, a, b$ ) in a serial fashion on the CPU to be used in the parallel GPU kernel (Algorithm 3). The computationally more intense tasks (Algorithm 3) rely in each GPU thread accessing an entry of the  $Y_f$  array in parallel for each grid point of the domain. Local coordinates are used in computing the  $tmp$  scalar to update  $Y_f$  and write it back. This operation requires 2 memory accesses, one read, and one write per  $N_h$  harmonic. We can thus evaluate the minimal amount of memory  $n_m$  (in Bytes) needed to be transferred at each  $N_h$  harmonic  $n_m = 2n_x n_y n_z n_p$ . The effective memory throughput  $MTP_{\text{eff}}$  (in GB/s) is then obtained as following:

$$MTP_{\text{eff}} = \frac{n_m N_h}{n_t 1024^3}, \quad (22)$$

where  $N_h$  is the number of iterations over the harmonics and  $n_t$  is the algorithm execution time in seconds. We compare  $MTP_{\text{eff}}$  values obtained for the random field simulator for single (SP) and double (DP) precision executions (Figure 9). We realise a strong scaling on a single Nvidia Tesla V100 (PCIe) GPU on which memory copy values are about 720 GB/s (dotted line in Figure 9). We report

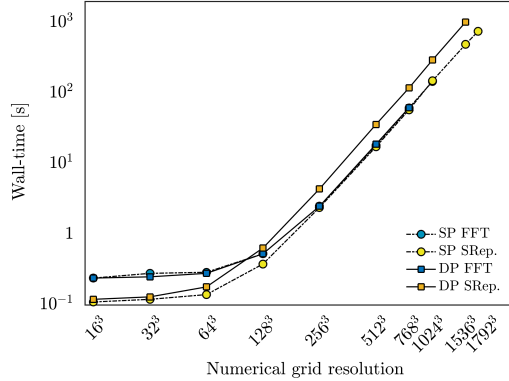


Figure 8: Wall-time [s] of the spectral representation (SRep) and FFT-based (FFT) 3-D random field simulator on a Nvidia Tesla Volta V100 GPU. We report the time to solution, varying the problem size from  $16^3$  to  $1792^3$  for both single precision (SP) and double precision (DP) arithmetic.

that the  $MTP_{\text{eff}}$  values saturated for both arithmetic precisions to 630 GB/s (88 % of memory copy) for computational domains containing more than 400 grid points in all 3 dimensions.

## 6. Applications

We utilise the random field simulator to investigate the effect of acoustic wave scattering as function of the anisotropy and statistical properties of the random medium. We combine the random field simulator with a 3-D finite-difference acoustic wave propagation simulator in the time domain (FDTD). We rely on a second order in time and fourth order in space discretisation of the non-split pressure formulation [39]. We include absorbing boundaries to avoid reflections of the wave at the artificial boundaries of the computational domain [40]. We use a cubic domain of  $L_x = L_y = L_z = 3000$  m, discretised on  $512^3$  grid points in 3-D. The width of the absorbing boundaries is of 400 m. The time dependent source of a vertically ascending planar wave is located on the  $z$ -plane of coordinate  $z = 0$ , at the bottom of the domain for the first configuration (Figure 10a) and at the top of the domain for the second configuration (Figure

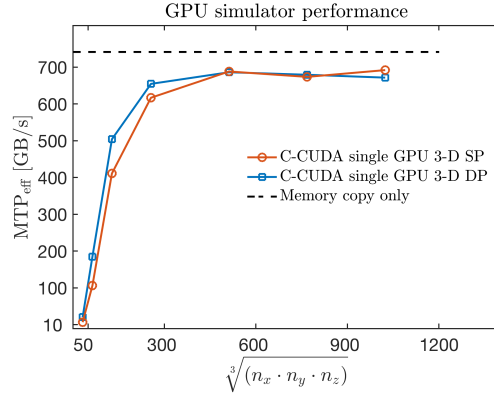


Figure 9: Effective memory throughput  $MTP_{\text{eff}}$  [GB/s] of the implemented 3-D random field simulator on a Nvidia Tesla Volta V100 GPU. We report a strong scaling, varying the problem size from  $32^3$  to  $1024^3$  for both single precision (SP) and double precision (DP) arithmetic.

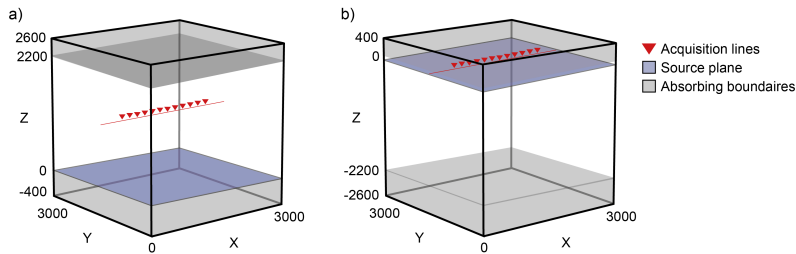


Figure 10: Initial conditions for the 3-D wave propagation model configuration including acquisition lines, source plane and absorbing boundaries. a) Plane wave coming from the bottom and acquisition line at mid-domain depth mimics a teleseismic wave observation in seismology. b) Acquisition line and source plane located on identical depths mimics a relevant configuration for multi-channel seismic reflection. Note that for better readability, the absorbing boundaries are only depicted in the vertical  $z$  direction but are implemented in a similar fashion in the the  $x$  and  $y$  directions.

10b). We simulate a forward planar wave propagation for 3 types of random  
 350 media displaying various correlation lengths and respecting identical box-size to  
 correlation lengths ratios as the random fields depicted in Figure 1.

We use a plane wave at the source plane to initiate the wavefield. The initial waveform  $s(t)$  corresponds to a Gaussian wavelet with the peak frequency ( $f_0$ )

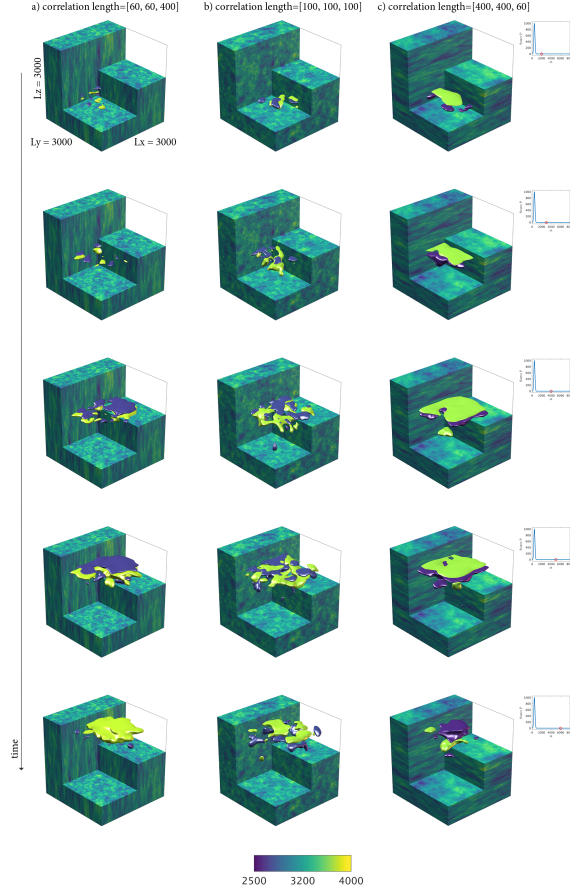


Figure 11: FDTD wave propagation in 3 distinct realisations of random media corresponding to the identical (a, b, c) ratios compared to box-size as depicted in Figure 1. The random fields are assigned to the seismic velocity within the forward simulations within the range of 2500 - 4000  $\text{ms}^{-1}$ . The planar wave source located at  $z = 0$  m (configuration as in Figure 10a) produces a scattered wavefield contoured for one positive and one negative pressure isosurface that evolves through time (vertical axis). The rightmost column displays the time dependent amplitude of the source perturbation. Animations related to this figure are available as supplementary material.

of 10 Hz:

$$s(t) = s_0 e^{-8f_0^2(t-t_0)^2}, \quad (23)$$

355 where  $s_0$  is the initial amplitude and  $t_0$  is the wave onset time.

We perform two forward simulations for each of the 3 utilised random fields; the first simulation is performed in homogeneous background velocity model ( $3200 \text{ ms}^{-1}$ ) while the second simulation includes the random velocity heterogeneities. This approach allows us to resolve the scattered wavefield (coda wave) by subtracting the background wavefield (direct wave) from the signal obtained in the randomly perturbed medium. We qualitatively report the contours of the pressure envelope (positive and negative iso-surfaces) for the coda wave evolving with time (vertical axis) through a vertically correlated (Figure 11a), isotropic (Figure 11b) and horizontally layered (Figure 11c) random medium.

365 In addition, we record the coda wave energy envelopes using a root mean square (RMS) ensemble average for 20 realisations for both investigated configuration; both at half the domain height ( $z = 1100 \text{ m}$ ) along the  $x$ -axis (Figure 12a, c, e) and at source ( $z = 0 \text{ m}$ ) location (Figure 12b, d, f) for the 3 distinct correlation length configurations (Figure 11a, b, c). We report a clear trend that energy envelopes are larger for the random medium displaying a minimal correlation length in the direction of the propagating wave compared to configurations where the correlation length is maximal along direction of the propagating wave, here the  $z$ -axis.

## 7. Discussion

375 We have considered geometrical anisotropy with an idea that the heterogeneity in the Earth can be represented in terms of continuous stochastic random heterogeneities. The analysis of seismic coda was particular fruitful to infer quantitative measures of the seismic scattering attenuation in the heterogeneous Earth [41]. Scattering of seismic waves in the crust assuming random heterogeneity has previously been extensively studied [e.g. 42]. Meschede and Romanowicz [27] using spectral Fourier method combined with the Karhunen-Loève expansion to construct non-stationary models on 3-D grids for global seismology applications. However, the size of covariance matrix and thus spa-



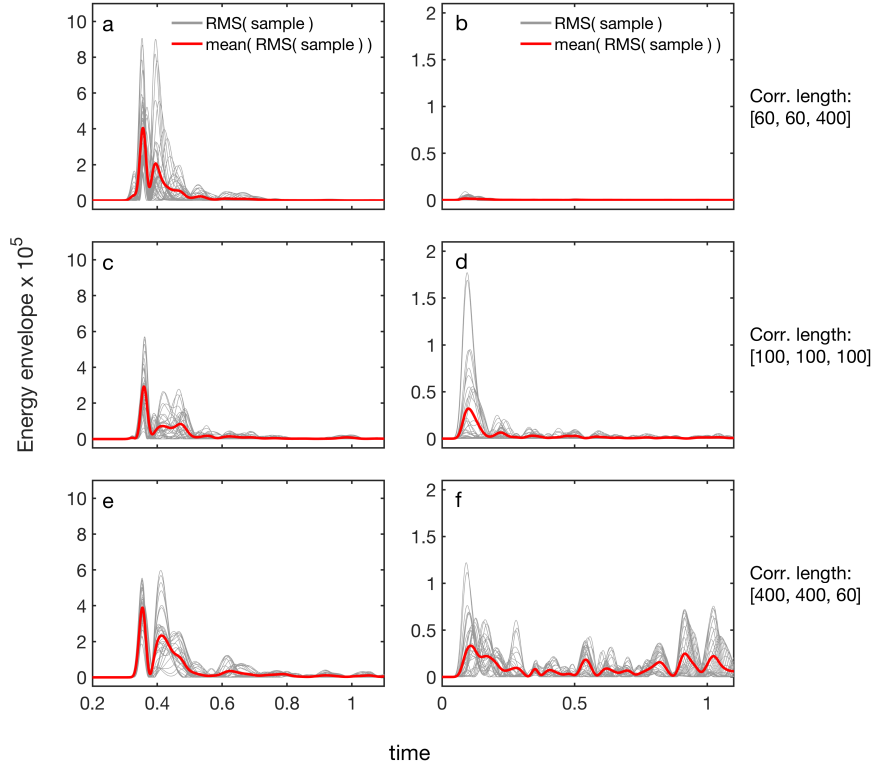


Figure 12: Coda wave energy envelopes for resulting from the planar acoustic wave propagation for random media with correlation lengths in  $[x, y, z]$  directions of (a-b)  $[60, 60, 400]$  m, (c-d)  $[100, 100, 100]$  m and (e-f)  $[400, 400, 60]$  m. The selected recording locations are at half the domain height (a, c, e) and at the source plane (b, d, f). The samples include RMS values of the full set of recorded waves for 20 realisations. The red line displays the RMS ensemble average. When applied to the studies of the Earth's mantle heterogeneities and P coda seismological observations, the model parameters upscale to be 300 km for the model vertical and lateral sizes, and the correlation lengths in  $[x, y, z]$  directions become (a)  $[6, 6, 40]$  km, (c)  $[10, 10, 10]$  km and (e)  $[40, 40, 6]$  km.

tial resolution was limited by the computer RAM in their simulations [27]. Our  
 385 method is free of this limitation and allows for an unstructured mesh and a  
 variable resolution across the model domain.

The interaction of seismic wave encountering a heterogeneity of comparable

or several times smaller size than the seismic wave length produces secondary scattered waves. The scattered wave train recorded after the first arrival of direct wave forms coda. Numerical simulations of coda are useful to examine the effects of small-scale structures on seismic waveforms. The scattering energy distribution and coda level in random media with different correlation lengths in radial and tangential direction depends on the incidence angle of the direct wave [43]. Thus, the coda energy variation in time can be explained by the energy partitioning between forward and backward scattered waves. The aspect ratio between the radial and tangential scales controls the strength of the backscattering. Scattered waves propagating backward or forward can be observed in receivers placed behind or in front of the scatterer. The forward scattered waves with small scattering angle arrive just after the direct wave. The numerical simulation of acoustic scattering is also important for better understanding the relative strengths of extrinsic (scattering) and intrinsic (absorption) seismic attenuation.

The first configuration (Figure 10a) including a plane wave coming from the bottom of the model mimics a teleseismic wave observation in earthquake seismology. The decay of seismic scattered coda waves represented by the energy envelopes provides information on statistical properties of the Earth's mantle heterogeneity [43]. In particular, backward scattering is stronger for planar heterogeneities (layered structures) whereas the vertically oriented structures (plumes, slabs) produce mainly forward scattering [43, 27]. Seismological studies suggest that the upper mantle contains strong heterogeneity of elastic properties at length scales from a few kilometres to tens of thousands kilometres [44]. The origin of small-scale heterogeneities in the mantle was attributed to both solid-state thermal convection facilitated by thermally activated creep processes and chemical anomalies forming as a result of partial melting of the mantle rocks [45]. Numerical simulations Schmalzl and Hansen [46] emphasised a scale-dependent efficiency of chemical mixing by thermal convection in the Earth's mantle. The sensitivity of teleseismic P-wave coda to the statistical properties (e.g. correlation function) of the mantle heterogeneity [47, 48, 44] can offer

useful constraints on the efficiency of mantle mixing across a broad range of  
420 scales and, thus, elucidate geochemical evolution of the Earth.

The second configuration (Figure 10b) is relevant for active seismic moni-  
toring of hydrocarbon reservoirs. Primary migration of hydrocarbons formed in  
the source rock occurs into a reservoir where oil and gas are accumulated if an  
overlying low-permeability seal (cap rock) exists. However, hydrocarbons may  
425 sometimes escape from the reservoirs through the cap rocks up to the surface  
by various mechanisms such as fracture, diffusion and porosity waves [9]. An  
active leakage process is potentially hazardous and must be considered in any  
prospect risk assessment. A hydrocarbon leakage can sometimes be directly  
observed as pockmarks and seepage plumes in the water column. Otherwise,  
430 3-D time-lapse seismic reflection data can be used for remote monitoring of the  
leakage processes that occur in the subsurface [49]. The standard processing  
of the seismic data includes the summation of coherent signal corresponding to  
the waves reflected and back-scattered in the sedimentary layer including both  
the stratified and 3-D fine-scale heterogeneities.

435 Zones of dimmed seismic amplitudes where the reflections from stratigraphic  
layers are significantly weaker than in adjacent areas are often encountered  
in seismic sections above leaking hydrocarbon reservoirs [49]. The amplitude  
dimming was attributed to a localised fluid or gas flow in the sediments based  
on a wealth of empirical data but the physical mechanisms behind are not well  
440 understood [49]. The relation between physical properties inside the zones of  
fluid flow and their expression in stacked multichannel reflection data is complex  
and non-unique. A realistic numerical experiment requires a 3-D geometry  
since the seismic energy scattered on small-scale pipe and vertical sheet seismic  
anomalies is different.

445 In Figure 12 the amplitudes of recorded back-scattered are significantly at-  
tenuated in the case of vertically oriented structures compare to the isotropic  
and planar horizontal heterogeneities despite that the mean bulk acoustic veloc-  
ity remains the same. This has an important implication for understanding the  
effective elastic properties of the sedimentary systems affected by localised pore

450 pressure changes and active fluid flow based on 3-D time-lapse seismic data.

## 8. Conclusion

Realistic numerical modelling of the Earth's spatial heterogeneity requires a length scale resolution across more than 4 orders of magnitude. In 3-D geometries, the computation is only feasible using parallel implementation of numerical algorithms. We suggest an efficient numerical method based on a randomisation technique [28] which allows to simulate the Earth's heterogeneity down to an arbitrary small scale assuming that the medium can be described as a random Gaussian field with a known correlation function or power spectrum. The GPU-accelerated numerical algorithm is described to simulate spatial realisations of a 3-D anisotropic homogeneous Gaussian random field in Cartesian coordinates. We showcase the ability of the approach to handle both exponential and Gaussian covariance functions. We confirm good performance results of the proposed method compared to a FFT-based approach, showing about one order of magnitude lower memory usage and comparable wall-times. Moreover, the proposed method to generate gaussian random fields based on spectral representation will scale linearly on distributed memory machines, as no communication is involved to reach the solution, which may not be the case for FFT-based approaches. The presented geophysical examples support the efficiency of the method and report the sensitivity of the recorded scattered seismic wavefields on the orientation and aspect ratio of the anisometric stochastic heterogeneity. These results can be applied to interpret seismic data across a wide range of scales: from a hydrocarbon reservoir to the upper mantle.

## References

- 475 [1] Minakov A, Keers H, Kolyukhin D, Tengedal HC. Acoustic waveform inversion for ocean turbulence. *Journal of Physical Oceanography* 2017;47(6):1473–91.

- [2] Kurbanmuradov O, Sabelfeld K, Koluhiin D. Stochastic Lagrangian models for two-particle motion in turbulent flows. Numerical results. Monte Carlo Methods and Applications 1997;3(3):199–224.
- 480 [3] Kraichnan RH. Convergents to turbulence functions. Journal of Fluid Mechanics 1970;41(1):189–217.
- [4] Kolyukhin D, Sabelfeld K. Stochastic flow simulation in 3D porous media. Monte Carlo Methods and Applications mcma 2005;11(1):15–37.
- [5] Sato H, Fehler MC, Maeda T. Seismic wave propagation and scattering in the heterogeneous earth; vol. 496. Springer; 2012.
- 485 [6] De Siena L, Del Pezzo E, Thomas C, Curtis A, Margerin L. Seismic energy envelopes in volcanic media: in need of boundary conditions. Geophysical Journal International 2013;195(2):1102–19.
- [7] Sabelfeld K, Loshchina N. Stochastic iterative projection methods for large linear systems. Monte Carlo Methods and Applications 2010;16(3-4):343–
- 490 59.
- [8] Christakos G. Random field models in earth sciences. Courier Corporation; 2012.
- [9] Räss L, Simon NS, Podladchikov YY. Spontaneous formation of fluid escape pipes from subsurface reservoirs. Scientific reports 2018;8(1):11116.
- 495 [10] Minakov A, Yarushina V, Faleide JJ, Krupnova N, Sakoulina T, Dergunov N, et al. Dyke emplacement and crustal structure within a continental large igneous province, northern Barents Sea. Geological Society, London, Special Publications 2018;460(1):371–95.
- 500 [11] Hansen TM, Mosegaard K, Cordua KS. Using geostatistics to describe complex a priori information for inverse problems. In: VIII International Geostatistics Congress; vol. 1. Mining Engineering Department, University of Chile; 2008, p. 329–38.

- [12] Mantoglou A, Wilson JL. The turning bands method for simulation of  
505 random fields using line generation by a spectral method. *Water Resources  
Research* 1982;18(5):1379–94.
- [13] Robin M, Gutjahr A, Sudicky E, Wilson J. Cross-correlated random field  
generation with the direct fourier transform method. *Water Resources  
Research* 1993;29(7):2385–97.
- [14] Le Ravalec M, Noetinger B, Hu LY. The FFT moving average (FFT-MA)  
510 generator: An efficient numerical method for generating and conditioning  
Gaussian simulations. *Mathematical Geology* 2000;32(6):701–23.
- [15] Kramer PR, Kurbanmuradov O, Sabelfeld K. Comparative analysis of  
multiscale Gaussian random field simulation algorithms. *Journal of Com-  
515 putational Physics* 2007;226(1):897–924.
- [16] Phoon K, Huang S, Quek S. Simulation of second-order processes using  
Karhunen–Loeve expansion. *Computers & structures* 2002;80(12):1049–60.
- [17] Phoon K, Huang H, Quek S. Simulation of strongly non-Gaussian pro-  
cesses using Karhunen–Loeve expansion. *Probabilistic Engineering Me-  
520 chanics* 2005;20(2):188–98.
- [18] Xiu D, Karniadakis GE. Modeling uncertainty in steady state diffusion  
problems via generalized polynomial chaos. *Computer methods in applied  
mechanics and engineering* 2002;191(43):4927–48.
- [19] Dowd PA, Xu C, Mardia KV, Fowell RJ. A comparison of methods  
525 for the stochastic simulation of rock fractures. *Mathematical Geology*  
2007;39(7):697–714.
- [20] Dowd P, Pardo-Igúzquiza E, Xu C. Plurigau: a computer program for sim-  
ulating spatial facies using the truncated plurigaussian method. *Computers  
& geosciences* 2003;29(2):123–41.

- 530 [21] Strebelle S. Conditional simulation of complex geological structures using multiple-point statistics. *Mathematical geology* 2002;34(1):1–21.
- [22] Tang Y, Atkinson PM, Wardrop NA, Zhang J. Multiple-point geostatistical simulation for post-processing a remotely sensed land cover classification. *Spatial Statistics* 2013;5:69–84.
- 535 [23] Deutsch CV, Cockerham PW. Practical considerations in the application of simulated annealing to stochastic simulation. *Mathematical Geology* 1994;26(1):67–82.
- [24] Tran N, Chen Z, Rahman S. Practical application of hybrid modelling to naturally fractured reservoirs. *Petroleum Science and Technology* 540 2007;25(10):1263–77.
- [25] Marcotte D, Allard D. Gibbs sampling on large lattice with GMRF. *Computers & Geosciences* 2018;111(June 2017):190–9. doi:10.1016/j.cageo.2017.11.012.
- [26] Holliger K, Levander AR, Goff JA. Stochastic modeling of the reflective lower crust: petrophysical and geological evidence from the Ivera 545 Zone (northern Italy). *Journal of Geophysical Research: Solid Earth* 1993;98(B7):11967–80.
- [27] Meschede M, Romanowicz B. Non-stationary spherical random media and their effect on long-period mantle waves. *Geophysical Supplements to the Monthly Notices of the Royal Astronomical Society* 2015;203(3):1605–25. 550
- [28] Sabelfeld KK. *Monte Carlo methods in boundary value problems*. Springer; 1991.
- [29] Rubinstein RY. *Simulation and the Monte Carlo Method*. 1st ed.; New York, NY, USA: John Wiley & Sons, Inc.; 1981. ISBN 0471089176.
- 555 [30] Eyink G, Goldenfeld N. Analogies between scaling in turbulence, field theory, and critical phenomena. *Physical Review E* 1994;50(6):4679.

- [31] Simonovski I, Cizelj L. Correlation length estimation in a polycrystalline material model. In: Proc. of the Int. Conf. Nuclear Energy for New Europe, Bled, Slovenia, 5–8 September. 2005, p. –.
- 560 [32] Tyaginov S, Vexler M, Sokolov N, Suturin S, Banshchikov A, Grasser T, et al. Determination of correlation length for thickness fluctuations in thin oxide and fluoride films. *Journal of Physics D: Applied Physics* 2009;42(11):115307.
- [33] Monin A, Yaglom A. *Statistical Fluid Mechanics of Turbulence*, Vol. 2. 1971.
- 565 [34] Katz RF, Weatherley SM. Consequences of mantle heterogeneity for melt extraction at mid-ocean ridges. *Earth and Planetary Science Letters* 2012;335:226–37.
- [35] Sava P, Poliannikov O. Interferometric imaging condition for wave-equation migration. *Geophysics* 2008;73(2):S47–61.
- 570 [36] Nvidia . *CUDA Programming guide*. 2018.
- [37] Box, G. E. P. and Muller, Mervin E. . A Note on the Generation of Random Normal Deviates. *Ann Math Statist* 1958;29(2):610–1.
- [38] Duretz T, Räss L, Podladchikov YY, Schmalholz SM. Resolving thermo-mechanical coupling in two and three dimensions: Spontaneous strain localisation owing to shear heating. *Geophysical Journal International* 2018;-.
- 575 [39] DeSanto JA. *Scalar wave theory*. Springer; 1992.
- [40] Moczo P, Kristek J, Galis M. *THE FINITE-DIFFERENCE MODELLING OF EARTHQUAKE MOTIONS: Waves and Ruptures*. Cambridge; 2014.
- 580 [41] Fehler M, Sato H. *Coda*. *Pure and applied geophysics* 2003;160(3-4):541–54.



- [42] Frankel A, Clayton RW. Finite difference simulations of seismic scattering: Implications for the propagation of short-period seismic waves in the crust and models of crustal heterogeneity. *Journal of Geophysical Research: Solid Earth* 1986;91(B6):6465–89.
- 585
- [43] Hong TK, Wu RS. Scattering of elastic waves in geometrically anisotropic random media and its implication to sounding of heterogeneity in the Earth’s deep interior. *Geophysical Journal International* 2005;163(1):324–38.
- [44] Mancinelli N, Shearer P, Liu Q. Constraints on the heterogeneity spectrum of Earth’s upper mantle. *Journal of Geophysical Research: Solid Earth* 2016;121(5):3703–21.
- 590
- [45] Khan A, Deschamps F. *The Earth’s Heterogeneous Mantle: A Geophysical, Geodynamical, and Geochemical Perspective*. Springer; 2015.
- [46] Schmalz J, Hansen U. Mixing the Earth’s mantle by thermal convection: A scale dependent phenomenon. *Geophysical research letters* 1994;21(11):987–90.
- 595
- [47] Hedlin MA, Shearer PM, Earle PS. Seismic evidence for small-scale heterogeneity throughout the Earth’s mantle. *Nature* 1997;387(6629):145.
- [48] Ricard Y, Durand S, Montagner JP, Chambat F. Is there seismic attenuation in the mantle? *Earth and Planetary Science Letters* 2014;388:257–64.
- 600
- [49] Løseth H, Gading M, Wensaas L. Hydrocarbon leakage interpreted on seismic data. *Marine and Petroleum Geology* 2009;26(7):1304–19.