Recap

Gabriele Tasca

2021

1 Overview

This is a recap of all the results obtained so far. The overall structure of the work is the following:

1. Simulate fractional Brownian surfaces, the simplest model of a self-affine surface.

2. Use the simulated surfaces to test and validate different methods of detecting the self-affine exponent.

3. Apply the methods to the glacier surfaces, and try to interpret the results by comparing them with the ones obtained with the simulated surfaces.

Some results are still included even if later results may make them less useful, for example the midpoint and spectral synthesis simulation methods, which are not too useful anymore after the exact method was implemented. They are left in because they may become useful again in later developments, such as when trying to build more complex models.

2 Simulation of Fractional Brownian Surfaces

2.1 Exact Simulation

Like any Gaussian process, a Fractional Brownian Field in $\mathbb{R}^n$ is completely specified by its mean vector μ and its covariance matrix Σ. In theory, an arbitrary Gaussian process can be simulated exactly by the following procedure [12, chapter 12.2]:

1. Specify $\mu = (\mu_1, ... \mu_n)^T$ and $\Sigma = \Sigma_{ij}, i, j = 1, ..., n$

2. Compute a square root A of Σ, so that $\Sigma = AA^T$

3. Generate a vector of independent standard normal variables:

$$Z = (Z_1, ..., Z_n)^T, Z_i = N(0, 1)$$

4. Output $X = \mu + AZ$.

By the Cholesky square root method, it is always possible to find the square root A for any covariance matrix Σ. Thus this method is applicable to processes with arbitrary μ and Σ. However, the Cholesky square root method is very computationally intensive for large matrices. If applied to a $n \times n$ square matrix, has a computational complexity of $O(n^3)$. If, like in the case of a FBS, the process under consideration takes values over a $n \times m$ grid in R^2, its covariance matrix will be a $mn \times mn$ square matrix, leading to a computational complexity of $O(n^3 m^3)$, which becomes intractable very quickly for the any computer.

In the case of a FBS, other methods have been proposed ([6], [12]), that allow for a faster computation of the square root by exploiting the speed of the Fast Fourier Transform algorithm.

To illustrate the method, it is useful to consider the case of a stationary Gaussian process on a regular grid $n \times n$ over the unit torus. The points on which the process takes value can be arranged either in a $n \times n$ matrix or in a vector of n^2 elements. The covariance matrix is a $n^2 \times n^2$ matrix. It can be divided into a $n \times n$ block structure, in each block is itself an $n \times n$ matrix and represents the correlation of each point with all other points of the grid.

Since we have considered a stationary process, the values of the correlation depend only of the difference between the grid indices. This means that the the values of each diagonal within each block are equal, as well as that the blocks that form a diagonal in the $n \times n$ block structure are equal to each other:

$$\Sigma = \begin{bmatrix} T_0 & T_1 & T_2 & \cdots & T_n \\ T_{-1} & T_0 & T_1 & \ddots & \vdots \\ T_{-2} & T_{-1} & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & T_1 \\ T_{-n} & \cdots & \cdots & T_{-1} & T_0 \end{bmatrix}, \tag{1}$$

in which each T_j block has the same structure internally. Such a matrix is said to have a block-Toeplitz structure with Toeplitz blocks.

In particular, due to the fact that in the unit torus the opposing sides in both directions are identified with each other, the covariance matrix has the further property that $T_n = T_{-1}$, $T_{n-1} = T_2$, and so on, so that the matrix is specified by its first row of blocks, and the ones following it are obtained by cyclic permutation of the first row. As before, the same is true for the inner structure of the individual blocks. This means that Σ is a block-circulant matrix with circulant blocks. The useful property of matrices of this kind is that they can be diagonalized by a combination of two Fourier transforms: in particular,

$$\Sigma = P^* \mathrm{diag}(\gamma) P, \tag{2}$$

where P is the Kronecker product of two $n \times n$ discrete Fourier transform matrices $F_{i,j} = \exp(-2\pi i i, j/n)/\sqrt{n}, i, j = 1, ..., n$, and $P*$ is its complex conjugate.

This means that $B = P^* \text{diag}(\sqrt{\gamma})$ is a square root of the covariance matrix Σ, which can be efficiently computed with the FFT algorithm and leads directly to the simulation of the process by the procedure described in the beginning of this section.

Application of this method to the case of fractional Brownian surfaces on non-toroidal grid requires the solution of two problems: first, that the covariance function of the fractional Brownian surface is not stationary, as it depends explicitly on s and t: if $X_H(t), t \in \mathbb{R}^2$ is the fBs process with Hurst parameter H,

$$\text{cov}(X_H(s), X_H(t)) = \|s\|^{2H} + \|t\|^{2H} + \|s - t\|^{2H}. \tag{3}$$

However, this problem is solved by simulating the stationary Gaussian process $\tilde{X}_H(t)$ with covariance

$$\text{cov}(\tilde{X}_H(s), \tilde{X}_H(t)) \equiv \rho(s, t) = c_0 + c_2\|s - t\|^2 - \|s - t\|^{2H}, \tag{4}$$

from which a process $X_H(t)$ with the desired covariance can be obtained from the combination

$$X_H(t) = \tilde{X}_H(t) - \tilde{X}_H(0) + \sqrt{c_2}\, t^T Z, \tag{5}$$

where $Z = (Z_1, Z_2)^T, Z_1, Z_2 = N(0, 1)$ is a vector of two independent identically distributed normal variables. Indeed, it can be checked that the covariance of the resulting process is given by [3].

The second problem is that on a non-toroidal grid the covariance matrix Σ cannot be block-circulant even in the case of a stationary process, but is only block-Toeplitz. This problem can be solved by embedding the Toeplitz matrix in a larger circulant matrix: this is obtained by embedding each of the $n \times n$ Toeplitz blocks into a larger $(2n-1) \times (2n-1)$ block constructed by "reflecting" the last $n - 1$ elements of each row and column into the newly added slots to the right and below the original block, and then filling the remaining spaces by extending the diagonals. The structure of resulting matrix is shown in figure [12.2][12].

From this larger block-circulant matrix $\tilde{\Sigma}$, the square root can be computed as described in the last paragraph, assuming that the larger matrix can also be treated as a covariance matrix. This means that $\tilde{\Sigma}$ must be non-negative definite, ensuring that its square root will be a real matrix. To guarantee this, a method introduced in [6] consists in simulating the process over a larger grid, with a covariance function that is equal to the desired one in the original grid and decays smoothly to zero: this is a sufficient condition for the existence of the embedding [5]. The parameters needed for ensuring this, and in particular the size of the extended grid, are different depending on whether H is lower or higher than 0.75: this leads to a higher computational cost in the case $H > 0.75$.

From this larger block-circulant matrix, the square root can be computed as described in the last paragraph, and selecting the appropriate sub-blocks results in a matrix that can be used to simulate the process $\tilde{X}_H(t)$. These steps can be organized in a way such that it is not necessary to hold in memory the whole covariance matrix, which holds highly redundant information because of the

Toeplitz property. The computational details on these methods can be found in chapters 12.2.2 and 12.4.4 of [12].

This method allows to sample fractional Brownian surfaces in an exact manner, without any kind of approximations or artifacts. This is very desirable for many applications, such as validating numerical analysis techniques that detect properties such as the fractal dimension and the self-affinity exponent. However, the requirements of this method in terms of both computational time and memory usage are still rather high: for an average computer with 4 gigabytes of memory, it can become troublesome to handle a grid of more than $n = 2^{10}/2^{11}$ points for each side, and for $n = 2^9$ the computation can take a time on the order of a second. This is due in significant part to the embedding procedures of the method, which mean that the matrices involved in the computation are larger than the $n \times n$ grid itself, by a factor between 2 and 4.

2.2 Approximate methods

For the reasons mentioned in the last paragraph, it is useful to rely on approximate methods for the simulation of fractional Brownian surfaces. The performance of approximate methods can improve significantly compared to the exact approach, both in terms of speed and memory usage. However, approximate methods can introduce significant artifacts in the resulting samplings, as well as bias in the results of some methods of analysis when applied to data generated in a specific way: for example, if one generates a sample of a fBs with a certain H_{gen} generated by spectral synthesis, analysis of the sample by spectral analysis might reveal an exponent H_{det} very close to H_{gen}, but the results of other methods may not be as accurate.

For this reason, it is useful to introduce and use more than one method of generation when validating methods.

2.3 Spectral Synthesis

It can be shown that a fractional Brownian surface of Hurst parameter H has a Fourier power spectrum S in the shape of a power law, with an exponent β related to H [3]:

$$S_H(k) \propto k^{-\beta}, \quad \beta = 2H + 2. \tag{6}$$

where $k = \sqrt{k_x^2 + k_y^2}$ is the wavevector modulus. This can be proven by calculating the power spectrum starting from the autocorrelation function, as mentioned in Chapter 2.4 of [3] for the case of one-dimensional fractional Brownian motion. The Fourier spectrum also can be linked directly to the Hausdorff dimension, as explained in Chapter 4.4 of [4].

This means it is possible to obtain an approximate sampling of fractional Brownian surfaces by first generating a spectrum of the shape [26], and then applying an inverse Fourier transform. In particular, we consider the case of a sampling of a fBs over an $n \times n$ grid $S_{i,j}$, labeled by indices $i, j(-n/2, ..., n/2))$, so that the point $i, j = (0, 0)$ corresponds to the center of the grid. The result

4

we want to obtain is a real function B_H defined on the grid points, so its discrete Fourier transform will be a complex function on the grid points that is symmetric in both i and j. After these considerations, the procedure is straightforward:

1. Generate a random complex field C_H over the sub-grid

$$\tilde{S}_{i,j}, \ i,j \in (0, ..., n/2). \tag{7}$$

   The field is given by
$$C = \text{Re}(C) + i \ \text{Im}(C), \tag{8}$$

   with both $\text{Re}(C)$ and $\text{Im}(C)$ uniformly distributed in $(0,1]$ (i.e. white noise fields), and the $\tilde{S}_{2,j}$ represents the positive quadrant of $S_{i.j}$, which holds all the independent components of the Fourier representation of B_H.

2. Calculate the power-law scaled spectrum D_H the complex field C by a normalized power law:

$$D_H = C \ (\gamma(H) + 1) \ r^{\gamma(H)}, \tag{9}$$

   where $r = \sqrt{i^2 + j^2}$ is the distance from the origin, and

$$\gamma(H) = \beta(H)/2 = -(H+1). \tag{10}$$

3. "Flip" the values of D_H over the remaining three quadrants of S_i, j, to obtain a symmetric complex field $\tilde{D}_H$ over the full grid S_i, j:

$$\tilde{D}_H(-i, -j) = D_H(i,j), \ i,j \in (0, ..., n/2). \tag{11}$$

4. Output the Fourier anti-transform of $\tilde{D}_H$:

$$B_H = \mathcal{F}^{-1}(\tilde{D}_H). \tag{12}$$

Although the results appear analogous to the results of exact fBs sampling, both visually as well as in most statistical tests, they must be considered as an approximation of the true process. The most clear fundamental difference is that the results of spectral synthesis B_H^{ss} are periodic, i.e. they take the same values on the opposite sides of the grid, in both directions. This is a natural consequence as the construction of B_H^{ss} as a Fourier sum of periodic sines and cosines, but it implies that the correlation of B_H^{ss} for any couple of opposite points is one, which is in general different from the value specified in [3] for the true process B_H.

   The relevance of remarks of this kind depends on the purpose that the simulation of the fBs process is meant to fulfill. In a work concerned with the study of fractional Brownian surfaces themselves, it would be certainly be important to take into account any kind of discrepancies between the simulated process and true fBs. However, since we are concerned fBs only as a generic and flexible model of a self-affine surface, we may equally well choose to assume any other

process as a model, provided it has the right self-affinity properties. As we will mention in the section about spectral regression, a power-law power spectrum can be directly linked to the fractal dimension of the surface, which is an indicator of the surface's self-affine behaviour. Thus, since the approximate fBs resulting from spectral synthesis has an exact power-law power spectrum, it can still be regarded as an exact model of a self-affine surface.

2.4 Midpoint Displacement

Another way to generate an approximate sampling of the fBs process is the midpoint displacement method [8, chapters 9.8 and 13.4]. This algorithm had been proposed in several slightly different forms, some of which are referred to as the successive random addition method, but all variations share the general idea. It was developed as a generalization of a method used to sample fractional Brownian motion in one dimension, defined on a sequence of points $S_i, i = 1, ..., N$ and taking values in $\mathbb{R}$. In the algorithm, N is taken to be a power of 2. In this simpler form, the method is a recursive procedure that works as follows:

1. Initialize the first, middle and last points of the sequence with a value taken at random from a Gaussian distribution, with variance $\left(\frac{1}{2}\right)^H$. Then, initialize all other points with the value 0.

2. Set a "generation number" g, starting at $g = 2$, to keep track of the recursion steps.

3. Consider a subset S_g of the points in the sequence, taken with spacing $n = N/2^g$ and always including the first and last points (for $g = 1$, the first and last points, plus the ones at one-quarter, half, three-quarters of the sequence).

4. By construction, the even points in S_g will still have the starting zero value, while the odd ones will already be set. Then, set the value of each even point as a linear interpolation of the two odd points neighboring it:

$$B_i = \frac{1}{2}(B_{i-1} + B_{i+1}) \tag{13}$$

5. Add a random Gaussian number, with variance $\left(\frac{1}{2}\right)^{Hg}$, to all points in S_g.

6. Return to step [3], and repeat the procedure with $g' = g + 1$. Continue until $2^g = N$, so that the last step applies to all points in the grid.

To apply the method to two-dimensional surfaces, the interpolation step can be adapted to include 2 steps. Starting from a grid with spacing $2^{N/g}$, first the center of all cells of the grid are interpolated from their diagonal neighbors, then the mid-points of the sides of the grid are interpolated from their orthogonal neighbors, completing the grid with spacing $2^{N/(g+1)}$ needed to proceed to the next steps.

2.4.1 Approximation quality

The results of the midpoint displacement method are compatible with the properties expected from the true fBs process, both at a visual level and in statistical tests of parameters such as the Hurst exponent and the fractal dimension. However, more study might be needed to understand theoretically the properties of the process resulting from this method.

2.5 Examples of the methods

Figures [15], [2] and [3] show some examples of the surfaces generated by the mentioned methods, for two different values of the self-affine exponent H. All surfaces are generated on a grid of 1024×1024 points.

For each value of H, the images produced by the three methods have a very similar visual appearance, the only major difference being that the output of the spectral synthesis method is periodic in both directions, since it is constructed as a Fourier antitransform, so that it is a sum of periodic functions. If necessary, however, a non-periodic sample can be obtained by simulating a bigger grid and extracting a submatrix of the desired size. In any case, we will see that the surfaces produced by the three methods have slightly differing statistical properties, which we will study in the next section by examining the different responses to various analysis methods aimed at detecting the self-affine exponent H. In turn, this will allow a better understanding of the assumptions and of the technical constraints that each method carries.

3 Detecting and analysing self-affine behaviour

In this section we will review some of the methods that can be applied when studying a real-world surface in order to detect a self-affine behaviour. The literature on this topic is vast and many methods have been proposed, although many of the published works have been concerned with the study of one-dimensional time series (real functions of one variable) and the extension to the two-dimensional case is not always straightforward. The reason for the abundance of different methods is that when analysing real data, there are many different quantities that can be studied as an estimator of the self-affine scaling exponent H, such as the power spectrum exponent, the fractal box-counting dimension, the divider dimension, the Hurst exponent and so on. While in most theoretical models of self-affine surfaces, such as fractional Brownian motion, these quantities are in precise relations with each other, so that they can be thought as functions of a single independent parameter H that qualifies the self-affine behaviour, in real data the relations may not hold, so that the values of the different estimators have to be considered as independent.

In addition, because the various methods differ so much both in the underlying principle and in numerical implementation, each of them presents an unique trade-off between accuracy, computational speed, and sensitivity to various factors such as a low number of data-points or the presence of gaps in the
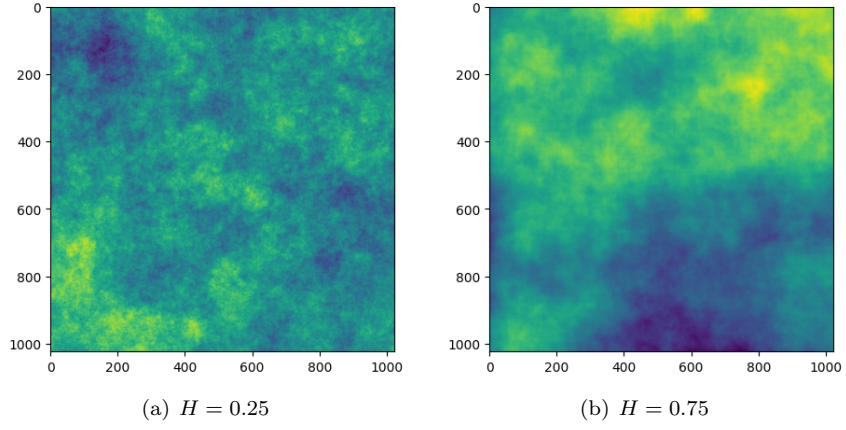
(a) $H = 0.25$ (b) $H = 0.75$

Figure 1: Examples of fBs generated by the exact method.

(a) $H = 0.25$ (b) $H = 0.75$
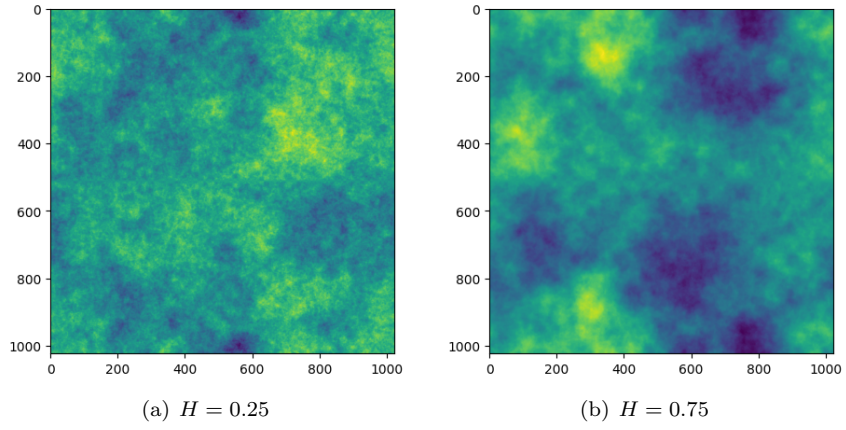
Figure 2: Examples of fBs generated by the spectral synthesis method.
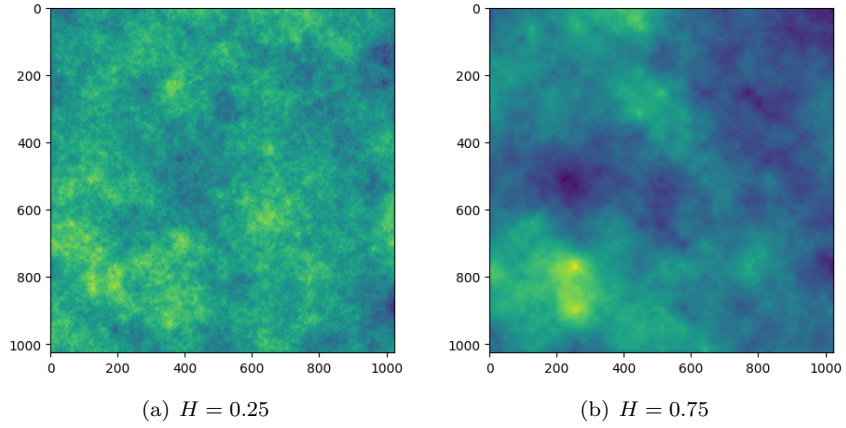

(a) $H = 0.25$ (b) $H = 0.75$

Figure 3: Examples of fBs generated by the midpoint displacement method.

8

data. For this reason, it is useful to have available as many methods as possible when studying a specific problem. In this work, we will attempt to give a characterization of the strengths and weaknesses of each method, with particular attention to the problems that arise in the study of glacier profiles, such as the fact that the surface is only defined over the irregular profile of the glacier and not on a full square grid.

3.1 Box counting

The first method we consider is the box-counting method, one of the oldest and most commonly employed method for the study of all kinds of fractal structures. The method provides a direct estimate of the box-counting dimension [**??**], which for most "normal" fractals is equivalent to the Hausdorff dimension, and can be linked to the self-affine exponent in the case of an ideal self-affine surface.

The data resulting from a measurement of a natural object is usually in the form of a discrete set of points F in $\mathbb{R}^d$, with $d \leq 3$ for points in three-dimensional physical space. In this case, the numerical application of the box-counting method usually involves dividing the space $\mathbb{R}^d$ in a grid of "boxes" of size δ, and counting the number $N(\delta)$ of boxes that contain at least one point. By repeating this analysis for many values of δ, one can study the variation of $N(\delta)$ as a function of δ. If a power-law behaviour is detected, i.e.

$$N(\delta) = c\,\delta^{-\beta}, \tag{14}$$

then β is the box-counting dimension of the set, and the estimation for the self-affine exponent is $H = d - \beta$.

In specific applications, the algorithm can be slightly modified according to the needs, for example by choosing a different box covering instead by the one provided by a square grid. Moreover, in the case of surfaces defined on a regular square grid, some other refinements can be made, due to the fact that it is reasonable to model the surface not just as the discrete set of points resulting from the measurements, but as the triangulated surface that one obtains by joining the points with small flat triangles. This idea is similar to the ones used in 3D computer graphics, in which the data relative to the vertices of an object is used to encode a surface made of triangles and/or quadrilaterals. This difference in interpretation can cause different estimates for $N(\delta)$, because in some cases the surface of a triangle can "clip" a box, so that it counts as intersecting the surface, even if it doesn't contain a point.

These observations have led to a number of slightly different forms of the box-counting method when applied to surfaces. Some of these are reviewed in [16], in the context of the study of rock fracture surfaces.

When working with data defined on a regular square grid, the available values of δ are integer multiples of the square grid size l. If one considers the points F_δ corresponding to a subset of the grid of size $\delta \times \delta$, the computationally faster way to count the number of boxes needed to cover them is to calculate the range $R(F_\delta) = \max(F_\delta) - \min(F_\delta)$ of the points within the grid subset. This seemingly

technical remark illustrates how the box-conting method is related to the R/S method [1], which studies the dependence range of a one-dimensional time series on the size of the window on which it is calculated. The R/S method is one of the oldest methods used to study the self-affine behaviour of one-dimensional time series, first introduced by Harold Edwin Hurst to study the water levels of the Nile river.

Figure [3.1] shows a study of the accuracy of the box-counting method. The plots report in the vertical axis the value of H detected by the method when applied to a fractional Brownian surface obtained by one of the three generation methods described in the last section, and the value of H used to generate them in the horizontal axis. Each data point is obtained as an average over a number $N = 20$ of independently generated surfaces, and error bar represents the standard deviation among these measurements. All surfaces are defined over a grid of 1024×1024 points, for a total of 1.048.576.

On surfaces generated by the exact method, the results of the box-counting method show a slight slope for values of $H_{gen} \leq 0.7$, causing an overestimation for low values of H_{gen} and a slight underestimation for $H \approx 0.7$. The different behavior for values above H_{gen} may be due to particularities in the exact generation method, which uses different embedding parameters below $H_{gen} < 0.75$. Overall, the method is fairly accurate in the region $H_{gen} < 0.5$, which is the one of interest in the study of glaciers and most natural surfaces.

On surfaces generated by the approximate methods, the slope of the H_{det} curve is higher, showing the limitations of these methods when trying to model all statistical features of true fractional Brownian surfaces.

3.2   Higuchi Method

The Higuchi method was introduced as method for studying the fractal properties of irregular time series [2]. It can be considered as an alternate method to estimate the box-counting dimension [14], but it is specialized to the case of continuous $\mathbb{R} \to \mathbb{R}$ curves. In this work, we apply a straightforward 2D generalization of the method, which applies to continuous surfaces represented as a $\mathbb{R}^2 \to \mathbb{R}$ continuous function, whose values are known on the points in the domain $\mathbb{R}^2$ corresponding to regular square grid. This extension of the method is analogous to the one described e.g. in [11].

Given a surface $F(i, j)$ defined on the grid $X_{i,j}$, with $i = 1, ..., N, j = 1, ..., M$, the method considers new surfaces constructed by taking points on the grid with a larger spacing k, and studying the variation of the area of the new surfaces depending on k. The precise steps of the method are as follows:
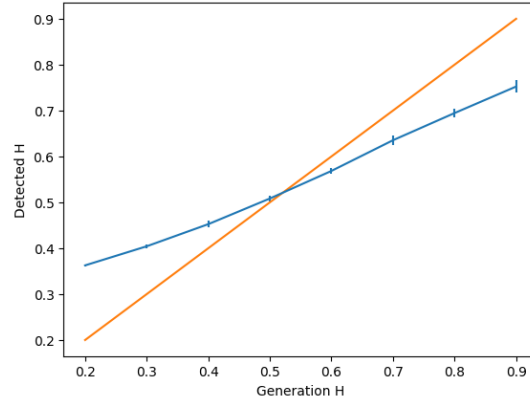
1. Set an integer scale parameter k, which assume values in the range $k = 1, ..., k_{max}$, where $k_{max} \approx N/15$.

2. Set two integer parameters m, n, which determines the shift of the new surface in the two directions, and will assume the values $n, m = 1, ..., k$.
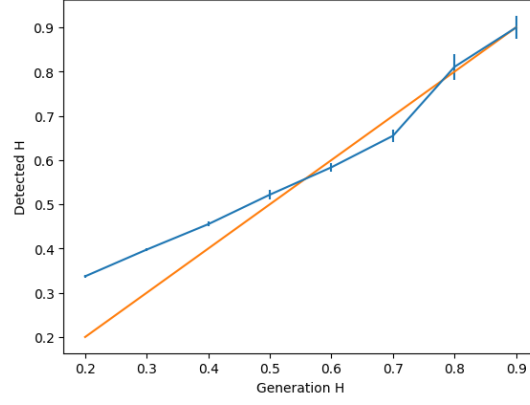
10

(a) Exact generation method.
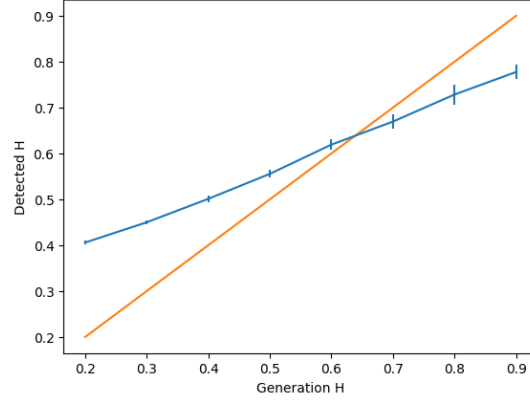
(b) Spectral generation method.

(c) Midpoint generation method.

Figure 4: Value of H detected by the box-counting method, for data generated by three methods, for various values of the generation H. Each point is obtained as an average over $N = 20$ generated surfaces.

11

3. Consider the new surface $F^{k,m,n}$ obtained by restricting the domain of F to the points $\tilde{X}_{v,w}$, with

$$v = n, \quad n+k, \quad n+2k, \quad ..., \quad n+\text{int}\left(\frac{N-n}{k}\right)\cdot k, \qquad (15)$$

$$w = m, \quad m+k, \quad m+2k, \quad ..., \quad m+\text{int}\left(\frac{M-m}{k}\right)\cdot k. \qquad (16)$$

$$(17)$$

4. For each square of the restricted grid $\tilde{X}_{v,w}$, let $F_{BL}^{k,m,n}$, $F_{BR}^{k,m,n}$, $F_{TL}^{k,m,n}$, and $F_{TR}^{k,m,n}$ be the bottom left, bottom right, top left, and top right points of the square. In other words, if a square is identified by the two values v, w, then $F_{BL} = F^{k,m,n}(v,w)$, $F_{TR} = F^{k,m}(v+k, w+k)$, and so on. Then, compute the following estimation of the area of the square:

$$A(k,m,n,v,w) = \text{abs}(F_{BL}^{k,m,n} - F_{BR}^{k,m,n})\cdot\text{abs}(F_{BL}^{k,m,n} - F_{TL}^{k,m,n})$$
$$+ \text{abs}(F_{TR}^{k,m,n} - F_{BR}^{k,m,n})\cdot\text{abs}(F_{TR}^{k,m,n} - F_{TL}^{k,m,n}). \quad (18)$$

Finally, sum over all squares to find

$$A(k,m,n) = \sum_{v,w} A(k,m,n,v,w). \qquad (19)$$

5. Repeat for all possible shifts $n, m = 1, ..., k$, and average the results to find a mean value $A(k)$.

6. Plot the values $A(k)$ as a function of k^2, so that they might be seen as the values of a new function $\tilde{A}(x) = A(k = x^{1/2})$. Study the dependence on the function $\tilde{A}$ on its parameter. As usual, if a power law is detected, e.g.

$$A(x) = c\,x^\beta, \qquad (20)$$

then the method has detected a fractal behaviour. The estimation for the box dimension is given by $D = 2 - \beta$, and the estimation for the self-affine exponent is $H = \beta + 1$.

Figure [3.2] shows the results of the Higuchi method on data generated by the three methods for a range of values of H_{gen}. Each data point is an average of the results obtained from $N = 10$ random surfaces, and the standard deviation is reported as an error bar.
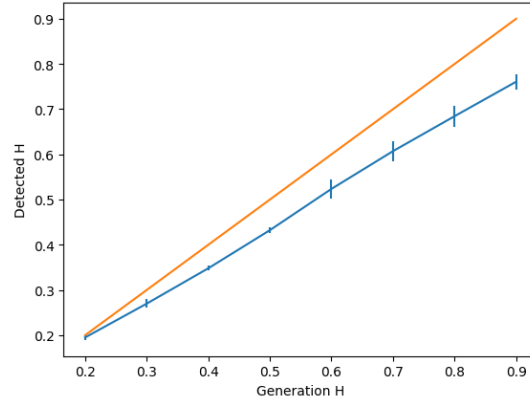
On data generated by the exact method, the Higuchi method shows an accuracy that depends differently on the value of H_{gen}, suggesting that different methods can carry slightly different responses on similar data even if the principles behind them are similar. On the data generated by the approximate methods, the Higuchi method shows a slope analogous to the one exhibited by the box-counting method, but with an overall slightly lower error.
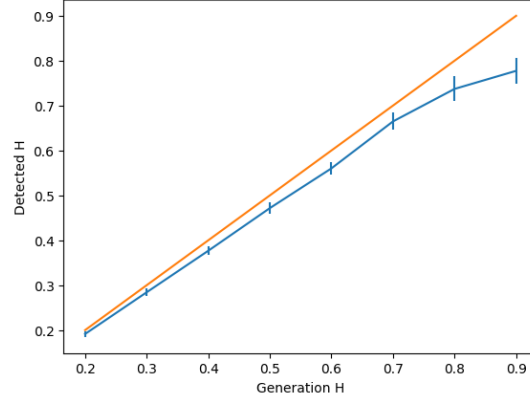
(a) Exact generation method.
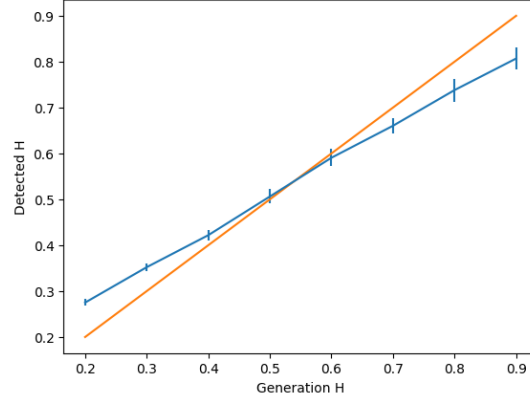
(b) Spectral generation method.

(c) Midpoint generation method.

Figure 5: Value of H detected by the Higuchi method, for data generated by three methods, for various values of the generation H. Each point is obtained as an average over $N = 10$ generated surfaces.

3.3 Detrended Fluctuation Analysis

The Detrended Fluctuation Analysis (DFA) method was introduced to study the long-range correlations in nucleotide sequences in DNA. It has since been widely employed as a measure of self-affinity, most often in the domain of one-dimensional time series. However, it can be extended to surfaces in $\mathbb{R}^3$ in a natural way, as shown for example in [7].

As usual, consider a surface defined as a two-dimensional array of real numbers $S(i,j), i = 1, ..., N, j = 1...M$, corresponding to the height value of the surface measured at the points of a regular square grid of spacing l. The procedure of the method is as following:

1. For a given box size δ which is an integer multiple of l, divide the points $S(i,j)$ in a grid of $\delta \times \delta$ square boxes. Each box is identified by two indices $\{w, v\}$, with $w = 1, ...\text{int}(N/\delta), w = 1, ...\text{int}(M/\delta)$.

2. For each box, consider the subset of points $\tilde{S}_{v,w}(i,j)$ contained in the box, and compute a least-squares fitting of the surface defined by the subset of points. For each grid point $\{i,j\}$ of the surface, call $u_\delta(i,j)$ the value of the fitting surface at that point. The simplest option is a plane:

$$u_{v,w}^\delta(i,j) = ai + bj + c, \tag{21}$$

but a quadratic function can be used as well:

$$u_{v,w}^\delta(i,j) = ai^2 + bj^2 + cij + di + ej + f. \tag{22}$$

In any case, the value of the parameters $a, b, ...$ can be easily estimated with standard least-squares fitting techniques. Note that u depends on δ, because for different values of δ the fitting is done over different portions of the surface.

3. For each point $\{i,j\}$ in the box, compute the difference between the real surface and the fitting surface:

$$\epsilon_{v,w}(i,j) = S_{v,w}(i,j) - u_{v,w}^\delta(i,j). \tag{23}$$

4. Compute the *detrended fluctuation* $F(\delta)$, by averaging the square of $\epsilon(i,j)$ over all points in all blocks, and taking the square root:

$$F(\delta) = \sqrt{\frac{1}{W_\delta\, V_\delta} \sum_{i,j} \frac{1}{\delta^2} \sum_{v,w} \epsilon_{v,w}(i,j)}, \tag{24}$$

where $W_\delta\, V_\delta = \text{int}(N/\delta)\text{int}(M/\delta)$ is the number of boxes of size δ.

5. Repeat the analysis for a wide range of values δ, and study the variation of $F(\delta)$ as a function of δ. If a power-law is detected, so that

$$F(\delta) = c\, \delta^H, \tag{25}$$

then H is the detected self-affine exponent of the surface.

Figure [3.3] shows the results of the DFA method on data generated by the three methods. In each case, the method presents a slight but distinct slope, seemingly uncorrelated by the ones seen in the previous methods. This suggests that when performing computational analyses of surface data, the relations between the different parameters studied by different methods do not hold strictly, and the various parameters should be all viewed as independent estimates.

3.4   Power Spectrum Regression

Studying the Fourier power spectrum of a self-affine is an effective and often very computationally efficient way to study its self-affinity characteristics. As was mentioned in Section [2.3], the power spectrum of fractional Brownian surfaces has a characteristic power-law dependence on the wave-vector modulus k:

$$S_H(k) \propto k^{-\beta}, \quad \beta = 2H + 2. \tag{26}$$

Relations of this kind can be obtained heuristically for more generic self-affine functions, by studying the properties of the autocorrelation function, as e.g in Chapter 11.2 of [4].

Based on these observations, the self-affine exponent can be estimated by simply computing the radially averaged Fourier power spectrum and fitting the resulting curve with a power law. The estimation for the Hurst exponent is then given by

$$H = -\frac{1}{2}(\beta + 2), \tag{27}$$

This method is conceptually straightforward, and can easily reach a high computational efficiency by relying on efficient software implementations of the FFT. However, there are some important technical precautions that must be taken in order to arrive to a reliable result. In fact, studying the power spectrum obtained by applying the FFT directly to the data can lead to significant artifacts and wrong results, and usage of different and imprecise conventions for the definitions of the Fourier transform has led to some confusion. These and other points are discussed in [13], in the context of rough surface topography.

In our case, the main issue is that the data are non-periodic, and thus applying the Fourier transform results in spectral leakage. In signal processing, non-periodicity is a very common issue that applies to virtually all measured signals, even the ones generated by periodic or stationary processes, because of course stationarity and periodicity are statistical properties of the underlying process which do not extend to any finite-size realization of it. From this point of view, however, it might seem that the non-periodicity of data such as ours is more fundamental than the one commonly encountered, because the data covers the whole glacier from start to finish, and is not taken as a subset of a larger object. Moreover, even the fractional Brownian surface process, which we use as an abstract model of a self-affine surface, is not stationary, as its covariance function [3] contains explicitly the two times t and s and not just the combination $t - s$. In fact, problems of this kind constitute concrete difficulties
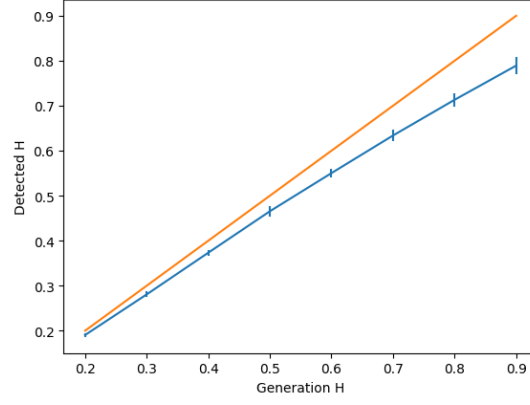
(a) Exact generation method.
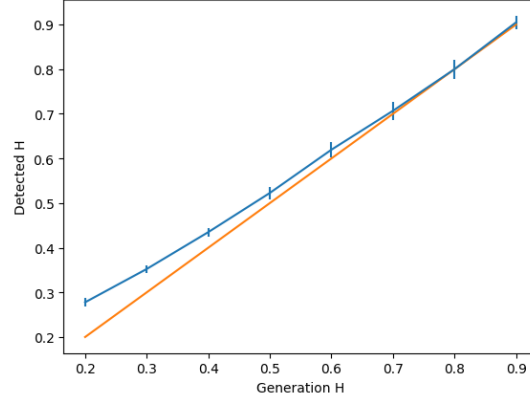
(b) Spectral generation method.
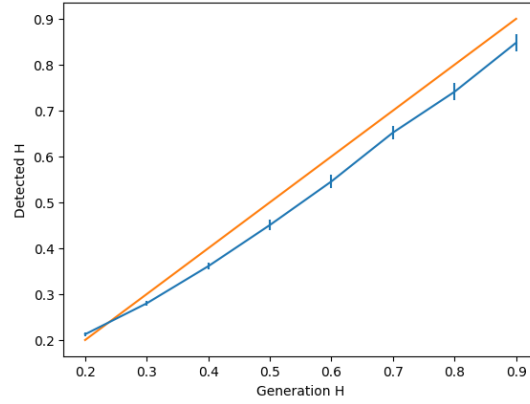
(c) Midpoint generation method.

Figure 6: Value of H detected by the DFA method, for data generated by three methods, for various values of the generation H. Each point is obtained as an average over $N = 20$ generated surfaces.

when studying rigorously the spectral properties of fractional Brownian motion processes, where in some cases are not possible to define at all [9]. In any case, when using Fourier transform methods, it is implicitly necessary to redefine the function under study over a periodic space.

This leakage does not only act as a source of imprecision in the measurement of spectral power, but can add a consistent systematic error to the evaluation of the power-law slope, and thus of the self-affine exponent. To see this, consider the case of a non-periodic surface defined on the points of a full square grid. Applying the Fourier transform to this data corresponds in applying a square windowing to the surface. Since the Fourier transform of a square window function has a $1/k$ asymptotic behavior, the power spectrum picks up a $1/k^2$ contribution. This causes the method to be highly ineffective for $H \gtrsim 0.5$.

This is a well known problem in many signal processing applications, and the most common solution is to apply a specific windowing function, which causes the spectral leakage to be distributed in a way that is less destructive to the spectral analysis. As suggested in [13], we employed a radial Hann window, which counters the leakage effectively without introducing anisotropy. On the other hand, applying windowing to a signal that is already periodic can cause the appearance of other kinds of artifacts, and in general will decrease the quality of the spectral analysis. This is the case of the data generated by the spectral synthesis method. In this case, the analysis is performed without any windowing.

Figure [3.4] shows the results of the power spectrum method. The method is fairly accurate, with a slight but constant underestimation with respect to the generation parameter. The magnitude of the underestimation is greater for the midpoint generation method.

3.5   Analysis of irregular profile surfaces

As was mentioned, the first difference one encounters while applying these methods to the study of real glacier surfaces is that, although a glacier surface can be seen as a function $\mathbb{R}^2 \to \mathbb{R}$, it is not defined over a square or rectangular subset of $\mathbb{R}^2$, but over the irregularly-shaped region in which the mountain is covered with ice. The perimeter of the ice-covered region is usually a fractal object itself [15].

The different analysis methods described in the last section need to be extended in different ways to be applied to irregularly-shaped surfaces, and their accuracy can be affected in different ways as well. To better understand the effect of this limitation of the results of the analyses, we generated synthetic glacier profiles by cutting irregular shapes from the square fractional Brownian surfaces, and we studied whether the analysis methods could still correctly detect the parameter H_{gen} that was used to generate them.

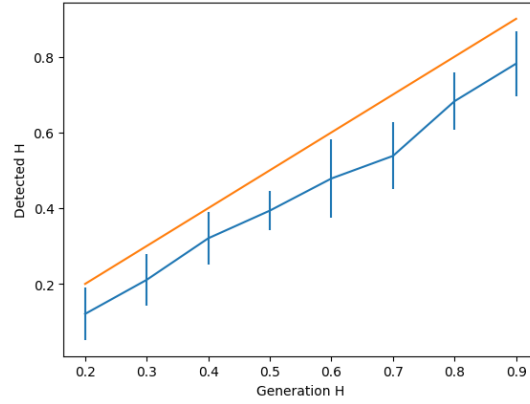To obtain the synthetic profile, the procedure is as follows:

1. Generate a fractional Brownian surface S, with self-affine parameter H_{gen}, defined on the full square grid $N \times N$, using one of the methods described
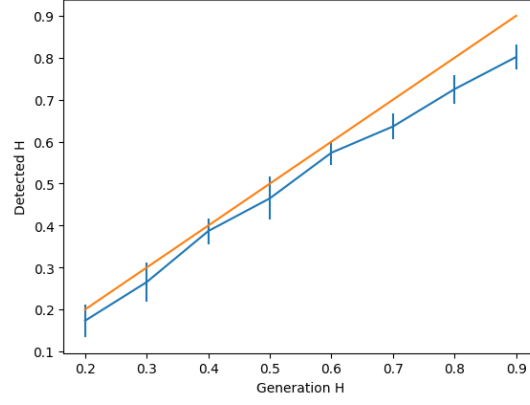
(a) Exact generation method.
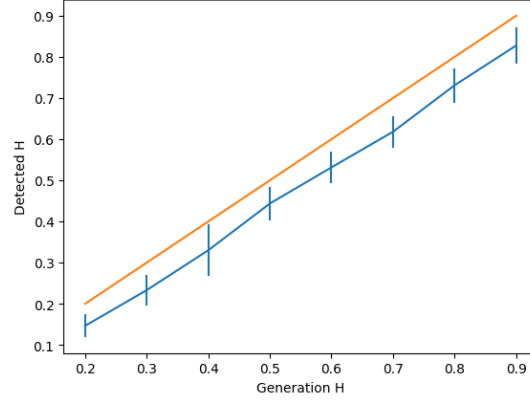
(b) Spectral generation method.

(c) Midpoint generation method.

Figure 7: Value of H detected by the power spectrum regression method, for data generated by three methods, for various values of the generation H. Each point is obtained as an average over $N = 20$ generated surfaces.

18

in section [2].

2. Generate a second fractional Brownian surface S_{prof} with self-affine parameter $H_{profile}$ with the same procedure as the first, to be used to obtain a "cutting mask".

3. Generate a third "surface" M that has height 1 where the values of the second surface S_{prof} are positive, and 0 where they are negative.

4. Output the product $R = S \cdot M$, discarding all points in which it equals zero.

By this method, the resulting surface is defined only inside of a contour line of S_{prof}. By using the contour line of a second surface, we can ensure that there is no correlation to the shape of the subset over which the resulting surface R is defined and the values of its height near the boundary. In general, the contour lines can have very different shapes, and usually generate a surface profile that is not composed of a single connected block. If more accurate simulation is necessary, connected blocks can be obtained by applying a flood fill algorithm to a point of the cutting mask and discarding the other parts.

The resulting surface depends on two parameters: the usual H_{gen}, which is the self-affine parameter of the surface, and H_{profile}, which characterizes the level of jaggedness of the contour line.
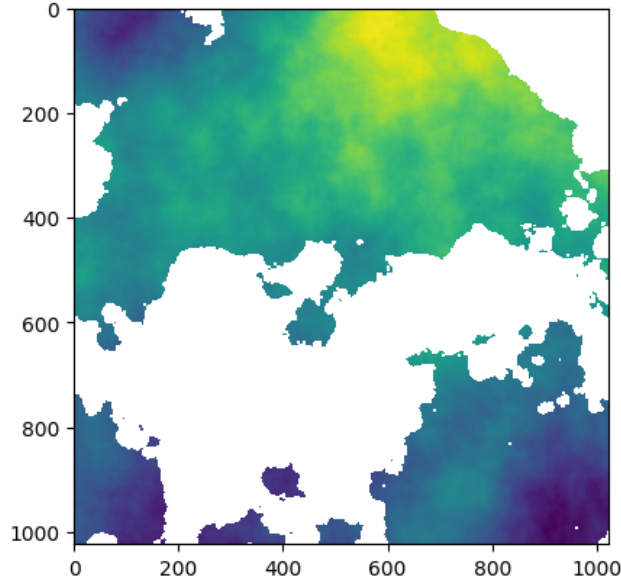

Figure 8: Example of a 1024×1024 synthetic glacier profile, with $H_{\mathrm{gen}} = 0.7$ and $H_{\mathrm{profile}} = 0.95$.

In the next sections, we discuss the modifications to the results of the analysis methods introduced in the last section needed to apply them to such surfaces, and report their results. We can anticipate that low values of H_{profile}, on the order of $H_{\text{profile}} \approx 0.4$ and lower, always result in very low accuracy of detection, because the surface becomes too fragmented for most methods to work. However it is immediately clear by looking at the data that the contours obtained with such low values are not representatives of the edges of real glaciers. For this reason, we report the results for more reasonable values of fig:profile-ex in the range $0.7 < H_{\text{profile}} < 0.95$.

3.6 Power Spectrum Regression

As was mentioned, even for a square surface, the discontinuity on the edge of the square is equivalent to applying a square window to the data, which affects very negatively the accuracy of analysis of the Fourier power spectrum. In the case of a square surface, this effect could be countered by the application of a smooth window such as the Hann window. In the case of irregularly-shaped images, similar discontinuities are present, but they are distributed irregularly in such a way that it is impossible to counter then by applying a ready-made window to the surface data. Typically, the power spectrum data becomes dominated by the square-window-like artifacts, which make it very difficult to obtain a reliable estimate for the "true" H of the surface. Thus, application of the power spectrum method on irregular surfaces is generally not useful.

If one wants to apply fruitfully any Fourier analysis methods to such surfaces, it could be beneficial to design a custom windowing function for each surface, that takes into account the shape of the boundary at least approximately. Another strategy to extract some information could be to apply the method to the largest connected square subset that can be obtained from the glacier's surface. In our case, given the availability of other methods which do not suffer from these problems, and because we want to develop a method of analysis that can be applied to different glaciers of varying shape and size, we opted to abandon the power spectrum method altogether.

Further discussion on the inadequacy of Fourier-transform based methods in the case of irregularly-shaped surface can be found in [10], where an alternative method based on the autocorrelation function is proposed.

3.7 Detrended Fluctuation Analysis

The DFA algorithm adapts rather naturally to the problem of irregular surfaces, because all data-points (δ, F) are obtained by as an average of the various square boxes of size $\delta \times \delta$ that the surface's domain can be covered with. It is then straightforward to limit the average over the boxes that do not overlap the boundary of the domain. The only bias originating from this approach is that as δ grows, the boxes of size δ provide a progressively more incomplete covering of the whole domain: for example, if a glacier has a "tongue" of width of order l, while the bulk of the glacier has a roughly circular shape with radius $R > l$,

20

then the tongue will not contribute at all to the values of F for $\delta > l$. However, this represents the fundamental fact that fluctuation over a scale larger than l is not well defined for the points of the tongue.

Figure [3.7] shows that the DFA method is still very accurate for low values of H_{gen}, but begins to give an underestimated result for higher values of H_{gen}.
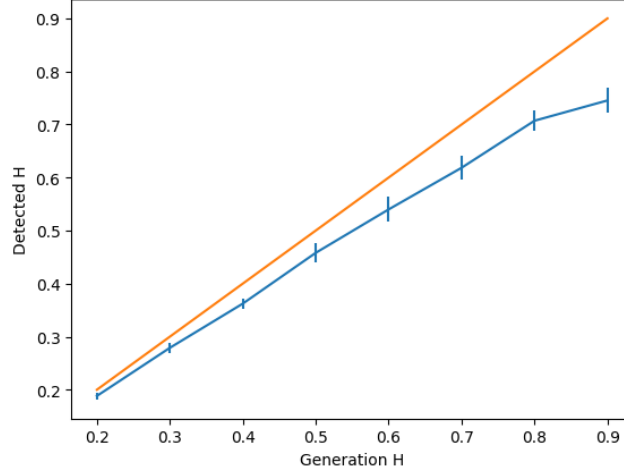

Figure 9: Value of H detected by the DFA method, for data generated by exact method, and boundary generated with $H_{\text{profile}} = 0.8$. Each point is obtained as an average over $N = 10$ generated surfaces.

3.8   Higuchi Method

In the Higuchi method, each point contributes to the estimation of the area by an amount which depends on the value of its neighboring points. As such, it is easy to skip a point in the estimation if its neighboring points lie outside of the surface's boundary. As in the case of DFA, some errors can appear when the linear size of the corrugations in the boundary are smaller than the analyzed scale δ. However, figure [3.8] shows that the response of the method picks up a downwards shift, corresponding to a consistent underestimation of the generation parameter.

3.9   Box-counting Method

The box-counting method is a general method that can be applied to any set of points in $\mathbb{R}^n$. In fact, the algorithm described in the last section can be applied to irregular surfaces with essentially no changes. However, the resulting value could not be compared to the ones obtained by the other methods. The reason for this is that the box-counting method treats the whole surface as a three-dimensional objects, by covering it with cubic boxes. However, the number
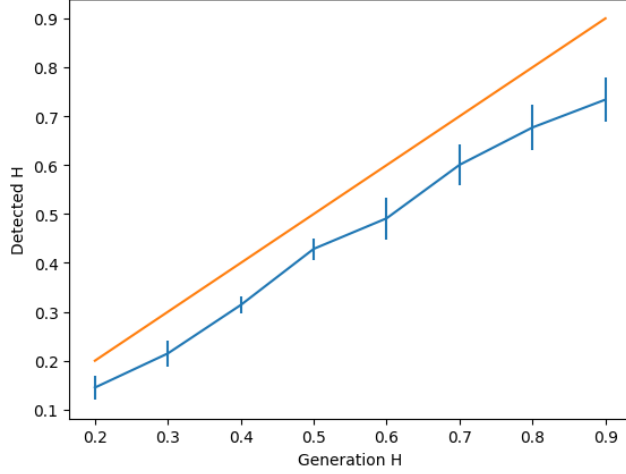
Figure 10: Value of H detected by the Higuchi method, for data generated by exact method, and boundary generated with $H_{\mathrm{profile}} = 0.8$. Each point is obtained as an average over $N = 10$ generated surfaces.

of boxes needed to cover the surface clearly depend both on the jaggedness of the surface and the jaggedness of the edge. Thus, in our case, the resulting exponent would be a combination of H_{gen} and H_{profile}, and couldn't be used as an estimation of H_{gen} alone. The fractal dimension of a glacier surface viewed as a fully three-dimensional object, as studied by the box-counting method, could be of interest on its own, but in a study of the properties of the surface, it is desirable to separate any effects caused by the shape of the boundary. It is nonetheless possible to adapt the box-counting method so that it provides information on the properties of the surface alone. As explained in section [3.1], the box-counting method works by fitting with a power law the number N of boxes needed to cover the surface as a function of the box size δ:

$$N(\delta) = c\,\delta^{-\beta}. \tag{28}$$

When analysing a full square surface, instead of the number $N(\delta)$, one might choose to track the volume of the boxes $V(\delta) = N(\delta) \cdot \delta^3$, the sum of the areas of their top surface $A(\delta) = N(\delta) \cdot \delta^2$, or any function $F_n(\delta) = N(\delta) \cdot \delta^n$. An analogous power law will hold:

$$F_n(\delta) = c\,\delta^{-\beta+n}, \tag{29}$$

and the desired β would be obtainable by subtracting n from the obtained exponent. In the case of an irregular surface, however, tracking the area $(n = 2)$ makes it easier to deal with the irregular shape. To see this, inscribe the irregular domain of the surface in a larger rectangle, and set to zero the height at the new points in which the height of the surface is not defined. When applying

22

the box-counting method to this new surface, the total area of the covering of boxes of side δ can be written as a sum of the ones that cover the real surface, and the ones that cover the flat zero-valued extension. However, the area of the covering of the flat part is constant and independent of δ, since it is flat. Thus, if the real surface would have followed a power law, the area of the covering of the new surface can be written as:

$$A(\delta) = c\, \delta^{-\beta+2} + K, \;\; K = \text{const.} \tag{30}$$

This expression can be fit numerically, and the real value of the self-affine parameter can be easily obtained from the detected exponent.

Even with this adjustment, the box counting method does not provide a very accurate measurement for the case of an irregular surface, as is shown in Fig. [3.9]. The accuracy is low, as seen by the wide error bars, and the measured H is consistent with the generation H_{gen} only for $H_{\text{gen}} > 0.5$.
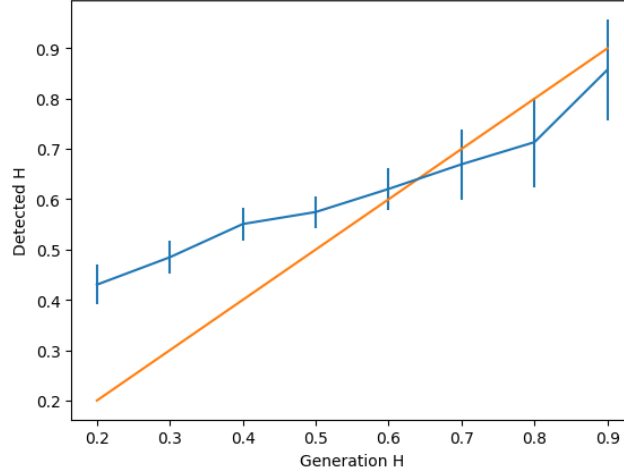

Figure 11: Value of H detected by the box-counting method, for data generated by exact method, and boundary generated with $H_{\text{profile}} = 0.8$. Each point is obtained as an average over $N = 10$ generated surfaces.

4   Analysis of Real Glaciers

4.1   Real and Projected Surface

Given the data describing a glacier surface at a certain resolution, it is possible to extend the discrete-point representation into an ideal surface by triangulation, i.e. taking the points to represent the vertices of a set of triangles that then form a continuous surface over the whole domain of the measurement. In the

case of a regular square grid, every cell defined by four vertices may be covered by two triangles, by choosing arbitrarily one of the two diagonals as an edge.

The area of the resulting surface can be computed as the sum of the areas of all the triangles. It is of course improper to refer to this area of the "real" area of the glacier, since it ignores all details of scale less than the resolution of the measurement. However, the ratio between this area and the area of the surface projected on its mean plane can be taken as a first indicator of the roughness of the glacier surface.

Projecting on the mean plane removes the contribution to the ratio given by the overall slope of the surface. Without this step, a perfectly smooth but very slanted glacier would still have a high ratio. This correction would not work in the case of a glacier whose surface cannot be accurately fitted by a single plane, such as a glacier that occupies both sides of a mountain ridge and has a different mean slope on each side. To eliminate these kind of effect, more sophisticated tools will be used in the next sections. Moreover, subtracting the mean plane from the surface will allow the ratio to assume values less than unity in some cases, because the projection is not orthogonal.

Figure [4.1] shows a histogram of the values of the area ratio for all the glaciers under study. The average value is $R = 1.0662$, with a standard deviation of 0.0511.
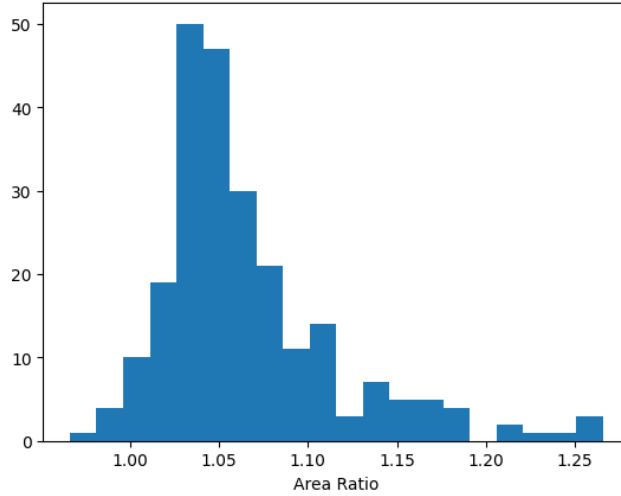

Figure 12: Histogram of the ratio between the area of the triangulated surface and the flat surface, obtained by projecting vertically on its mean plane.

4.2   Autocorrelation

The study of the autocorrelation function can often provide useful information about the self-affine behavior of a surface F. For an ideal self-affine surface,

the autocorrelation function is linked to the Fourier power spectrum by the Wiener–Khinchin theorem, which implies that it also follows a power law with an exponent that can be traced back to the self-affine parameter H:

$$C(r) = \langle F(x_1)F(x_2) \rangle = r^{-\gamma}. \tag{31}$$

where the average is taken over all couples of points x_1, x_2 so that $\text{dist}(x_1, x_2) = r$.

The autocorrelation function is usually estimated by very fast FFT methods that exploit the Wiener–Khinchin theorem. However, such methods are subjects to the edge effects and windowing artifacts mentioned in sections [3.4] and [3.5]. Thus, it is preferable to estimate the correlation by space-domain methods [10]. In our case, for each value of r, we simply sampled at random $N = 100000$ couples of points with distance r between each other, and averaged the product of the surface height over all couples. We verified the accuracy of the method by comparing the results to the ones of the usual FFT calculation on test surfaces designed to be FFT-friendly, and found that the space-domain method si able to reproduce the results in these cases, although it is less time-efficient. On both the real glacier surfaces and the synthetic ones described in section [??], however, we found that the FFT method introduces significant bias. Figure [13] shows the results of the application on the Forni glacier, the biggest one of the batch. For low r, the FFT method shows a deviation visually analogous to the square-windowing and discontinuity effects in the Fourier power spectrum. (Note that figure [13] is in linear scale, to better highlight the deviations).
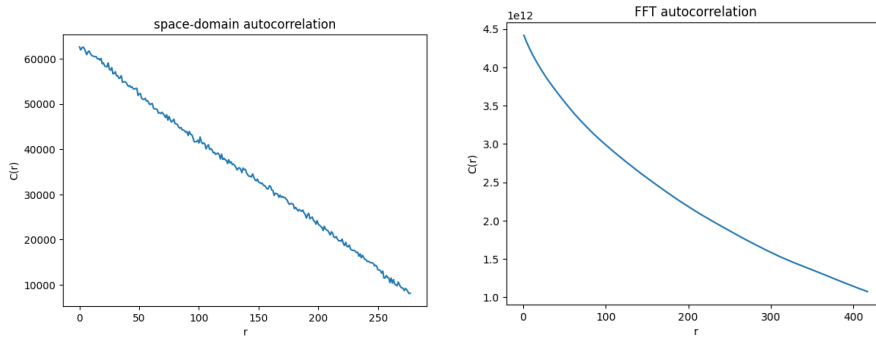

Figure 13: Radially averaged autocorrelation of the Forni glacier surface, calculated by the space domain and FFT methods. Both plots are in linear scale.

In our case, a first look at the plots of the radially averaged autocorrelation function does not reveal any information about the self affine behavior, because the plots do not show a power-law dependence, as shown in figure [14]. Figure [13] shows that on a linear scale, the plot is very close to being linear. The shape of the log-scale plot [14] is qualitatively similar to the one found in [10], in the context of a study of the surface of mouse fibroblast cells with somewhat similarly shaped data. As cited there, [16] provides a discussion on how such

deviations from the expected power law behavior may be caused by the discrete sampling of the surface. Using this information, [10] proposes a method which is able to extract reliable information about the self-affine exponent by studying the behavior of the autocorrelation at low r.
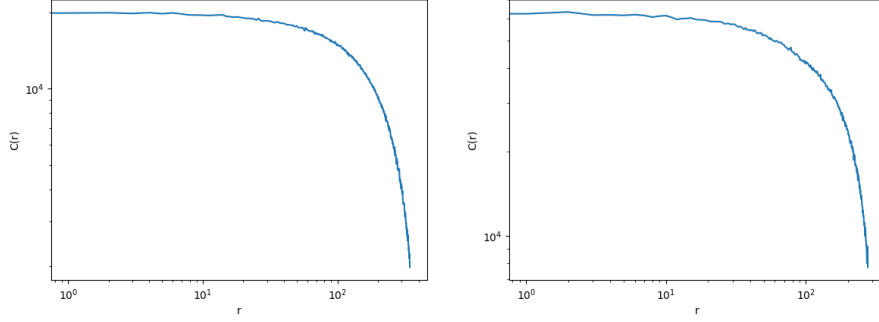

Figure 14: Radially averaged autocorrelation of the Adamello and Forni glacier surfaces. Both plots are in logarithmic scale.

Calculating the autocorrelation of a discretely sampled flat plane returns a result similar to the ones in Figures [14] and [13]. This offers another possible contribution to the deviations from the power law curve: at some scale, the surface may be behaving in a non-fractal way analogous to a combination of many flat surface patches. This explanation seems to be consistent with some results we will mention in the next section, in which the surface appears to have a self-affine exponent close to $H = 1$ for larger scales.

## 4.3   Self-Affine Exponent Analysis

In this sections, we apply the methods studied in section 3. For many of the smaller glaciers, the number of data points is quite small, and more work will be needed in the future to make sure that the methods produce reliable results even with a low amount of points. For now, we focused on the analysis of the larger glaciers, in order to try to understand the qualitative features of the results. Figure [4.3] shows the results of the applications of the box-counting, Higuchi and DFA methods to the Adamello glacier, one of the largest of the batch. These results are qualitatively similar to most of the larger glaciers. The plots show the dependence of the method's estimation parameter (area of the box covering for the box-counting method, area of the reduced image for the Higuchi method, and the fluctuation function for the DFA method) on the analysis scale. In all three cases, a single power law does not model accurately the dependence. In the DFA and box-counting result, the data appears to follow two separate power laws over two ranges of scales, separated by a cutoff scale, whose value is roughly consistent across the methods. The results of the Higuchi method seem consistent with this hypothesis, although this last method shows a smoother curve instead of an abrupt change of slope.

The plots show in green and red two power laws that fit the small-scale and large-scale segments of the data, selected empirically. The exponents of the small-scale and large-scale power laws are not consistent across the different methods, especially in the large-scale one. This may be due to numerical issues of the methods: in particular, we found that the different methods lead to very different estimates when applied to almost-flat surfaces in which the fractal behavior is small or nonexistent.

5   Next steps

Here is a recap of the possible next steps.

1. Apply the method described in [10], to try to estimate H starting from the space-domain autocorrelation data.

2. Continue experimenting and validating the methods to better understand the results in Figure [4.3], and to eventually obtain consistent estimates for at least one H across different methods.

3. Try to build a model for the two-slope behavior. For example, sum together two fractional Brownian surfaces with different H, or alter the midpoint or spectral synthesis generation methods so that it uses a different H over different ranges of scales.

4. Successively, test and validate the methods when applied to small glaciers, so that a reliable value of H can be extracted from all (or most) glaciers, opening the way to batch-processing of the resulting H values, to study distributions, correlations with other parameters, etc.
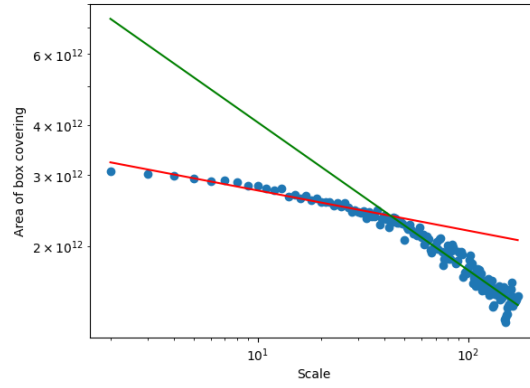
(a) Box-counting method.
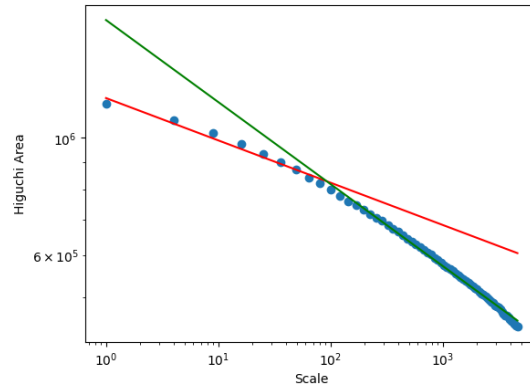
(b) Higuchi method.
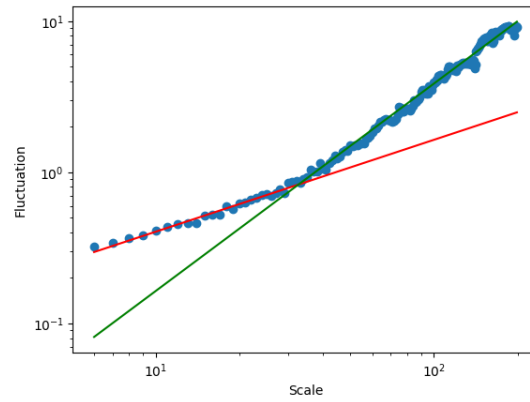
(c) DFA method.

Figure 15: Data obtained by applying three methods of analysis to the Adamello glacier.

References

[1] Benoit B. Mandelbrot and James R. Wallis. "Computer Experiments with Fractional Gaussian Noises: Part 2, Rescaled Ranges and Spectra". In: *Water Resources Research* 5.1 (1969), pp. 242–259. DOI: `https://doi.org/10.1029/WR005i001p00242`. eprint: `https://agupubs.onlinelibrary.wiley.com/doi/pdf/10.1029/WR005i001p00242`. URL: `https://agupubs.onlinelibrary.wiley.com/doi/abs/10.1029/WR005i001p00242`.

[2] T. Higuchi. "Approach to an irregular time series on the basis of the fractal theory". In: *Physica D: Nonlinear Phenomena* 31.2 (1988), pp. 277–283. ISSN: 0167-2789. DOI: `https://doi.org/10.1016/0167-2789(88)90081-4`. URL: `https://www.sciencedirect.com/science/article/pii/0167278988900814`.

[3] Heinz-Otto Peitgen and Dietmar Saupe, eds. *The Science of Fractal Images*. Berlin, Heidelberg: Springer-Verlag, 1988. ISBN: 0387966080.

[4] Kenneth Falconer. *Fractal geometry - mathematical foundations and applications*. Wiley, 1990, pp. I–XXII, 1–288. ISBN: 978-0-471-92287-2.

[5] C. R. Dietrich and G. N. Newsam. "Fast and Exact Simulation of Stationary Gaussian Processes through Circulant Embedding of the Covariance Matrix". In: *SIAM Journal on Scientific Computing* 18.4 (1997), pp. 1088–1107. DOI: `10.1137/S1064827592240555`. eprint: `https://doi.org/10.1137/S1064827592240555`. URL: `https://doi.org/10.1137/S1064827592240555`.

[6] Michael L Stein. "Fast and Exact Simulation of Fractional Brownian Surfaces". In: *Journal of Computational and Graphical Statistics* 11.3 (2002), pp. 587–599. DOI: `10.1198/106186002466`. eprint: `https://doi.org/10.1198/106186002466`. URL: `https://doi.org/10.1198/106186002466`.

[7] Gao-Feng Gu and Wei-Xing Zhou. "Detrended fluctuation analysis for fractals and multifractals in higher dimensions". In: *Phys. Rev. E* 74 (6 Dec. 2006), p. 061104. DOI: `10.1103/PhysRevE.74.061104`. URL: `https://link.aps.org/doi/10.1103/PhysRevE.74.061104`.

[8] J. Feder. *Fractals*. Physics of Solids and Liquids. Springer US, 2013. ISBN: 9781489921246.

[9] E. Kuleshov and B. Grudin. "Spectral density of a fractional Brownian process". In: *Optoelectronics, Instrumentation and Data Processing* 49 (May 2013). DOI: `10.3103/S8756699013030035`.

[10] Douglas H. MacDonald et al. "Autocorrelation method for fractal analysis in nonrectangular image domains". In: *Opt. Lett.* 38.21 (Nov. 2013), pp. 4477–4479. DOI: `10.1364/OL.38.004477`. URL: `http://ol.osa.org/abstract.cfm?URI=ol-38-21-4477`.

[11] Sladjana Spasić. "On 2D generalization of Higuchi's fractal dimension". In: *Chaos, Solitons Fractals* 69 (2014), pp. 179–187. ISSN: 0960-0779. DOI: https://doi.org/10.1016/j.chaos.2014.09.015. URL: https://www.sciencedirect.com/science/article/pii/S0960077914001696.

[12] Dirk P. Kroese and Zdravko I. Botev. *Spatial Process Simulation*. Ed. by Volker Schmidt. Cham: Springer International Publishing, 2015, pp. 369–404. ISBN: 978-3-319-10064-7. DOI: 10.1007/978-3-319-10064-7_12. URL: https://doi.org/10.1007/978-3-319-10064-7_12.

[13] Tevis D B Jacobs, Till Junge, and Lars Pastewka. "Quantitative characterization of surface topography using spectral analysis". In: *Surface Topography: Metrology and Properties* 5.1 (Jan. 2017), p. 013001. DOI: 10.1088/2051-672x/aa51f8. URL: https://doi.org/10.1088/2051-672x/aa51f8.

[14] Lukas Liehr and Peter Massopust. "On the mathematical validity of the Higuchi method". In: *Physica D: Nonlinear Phenomena* 402 (2020), p. 132265. ISSN: 0167-2789. DOI: https://doi.org/10.1016/j.physd.2019.132265. URL: https://www.sciencedirect.com/science/article/pii/S0167278919303859.

[15] Marina Carpineti et al. "Multifractal analysis of glaciers in the Lombardy region of the Italian Alps". In: *Journal of Physics: Complexity* 2.2 (Feb. 2021), p. 025003. DOI: 10.1088/2632-072x/abd3ae. URL: https://doi.org/10.1088/2632-072x/abd3ae.

[16] Mingyang Wu et al. "Improved box-counting methods to directly estimate the fractal dimension of a rough surface". In: *Measurement* 177 (2021), p. 109303. ISSN: 0263-2241. DOI: https://doi.org/10.1016/j.measurement.2021.109303. URL: https://www.sciencedirect.com/science/article/pii/S0263224121003067.