

Research Article

Efficiently Implementing the Maximum Likelihood Estimator for Hurst Exponent

Yen-Ching Chang^{1,2}

¹ Department of Medical Informatics, Chung Shan Medical University, No. 110, Section 1, Jianguo North Road, Taichung 40201, Taiwan

² Department of Medical Imaging, Chung Shan Medical University Hospital, No. 110, Section 1, Jianguo North Road, Taichung 40201, Taiwan

Correspondence should be addressed to Yen-Ching Chang; nicholas@csmu.edu.tw

Received 13 February 2014; Accepted 28 March 2014; Published 30 April 2014

Academic Editor: Matjaz Perc

Copyright © 2014 Yen-Ching Chang. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

This paper aims to efficiently implement the maximum likelihood estimator (MLE) for Hurst exponent, a vital parameter embedded in the process of fractional Brownian motion (FBM) or fractional Gaussian noise (FGN), via a combination of the Levinson algorithm and Cholesky decomposition. Many natural and biomedical signals can often be modeled as one of these two processes. It is necessary for users to estimate the Hurst exponent to differentiate one physical signal from another. Among all estimators for estimating the Hurst exponent, the maximum likelihood estimator (MLE) is optimal, whereas its computational cost is also the highest. Consequently, a faster but slightly less accurate estimator is often adopted. Analysis discovers that the combination of the Levinson algorithm and Cholesky decomposition can avoid storing any matrix and performing any matrix multiplication and thus save a great deal of computer memory and computational time. In addition, the first proposed MLE for the Hurst exponent was based on the assumptions that the mean is known as zero and the variance is unknown. In this paper, all four possible situations are considered: known mean, unknown mean, known variance, and unknown variance. Experimental results show that the MLE through efficiently implementing numerical computation can greatly enhance the computational performance.

1. Introduction

Signals of nature [1–6], medicine [7–14], business [15–18], and society [19–22] usually appear to be a strong long-term correlation. These signals can be differentiated by only one indicator, fractal dimension or Hurst exponent; therefore, many researchers are attracted to the study of how to estimate fractal dimension or Hurst exponent. In order to analyze the characteristics of fractal signals, users can determine the fractal dimension (D). Among estimators, the box-counting technique [23–27] is a direct nonmodeling method. In general, engineers are fond of adopting indirect modeling methods like fractional Brownian motion (FBM) or fractional Gaussian noise (FGN), because they are more meaningful than direct nonmodeling methods. FBM or FGN first estimates the Hurst exponent (H), a real number in $(0, 1)$, and then calculates the fractal dimension via the relation $D = 2 - H$ [28]. The Hurst exponent is the only one parameter

dominating the characteristics of FBM or FGN. FBM is a statistically self-similar nonstationary random process, which makes analysis difficult [29, 30], but the increment of FBM, FGN, is a strict-sense stationary process and has power spectral density (PSD) behaving asymptotically as f^{1-2H} [29, 31].

In real applications, signals must be sampled in advance; sampled FBM is called discrete-time fractional Brownian motion (DFBM), and sampled FGN is called discrete-time fractional Gaussian noise (DFGN), which has been proven to be a regular process [32]. Many natural and biomedical signals can be modeled as DFBM or DFGN [7–12, 33]. Among estimators, the maximum likelihood estimator (MLE) [29] provides the optimal accuracy; one of its approximate versions, called the Whittle estimator [34, 35], provides the second optimal accuracy. The aim of the Whittle estimator is to provide faster estimation with slight inaccuracy. Other

quick versions include the variance method [29, 31, 36], moving-average (MA) method [33], and autoregressive (AR) method [32, 37].

Although the accuracy of estimating the Hurst exponent by the MLE is the best, it is easy to induce computational problems and enormous computational expenditure. For example, evaluating the inverse of an autocovariance matrix may be numerically unstable, especially when H is close to 1. Under this situation, the autocovariance matrix of DFGN changes very slowly [35]. This problem will cause computational inaccuracy, leading to wrong explanations for physical signals of interest. On the other hand, the cost often makes users hesitate to apply the MLE to quick response systems, and thus the theoretical value of the MLE is generally much higher than its practical applications.

When taking a closer look at the structure of the autocovariance matrix, a combination of the Levinson algorithm [38] and Cholesky decomposition [39] can solve computational problems and reduce computational cost. Accordingly, users will be encouraged to adopt the MLE even in the quick response situations, and then the MLE has a better opportunity to become the first choice in the future, especially when computer speed continues to be increased up to a certain level.

When the MLE was first proposed by Lundahl et al. [29], the analysis and evaluation of the MLE were based on the assumptions that the mean of DFGN is zero and the variance is unknown. It is only applicable to physical signals modeled as DFBM, but not suitable for the model of DFGN. When signals are modeled as DFGN, it is easy for users to obtain wrong estimation results, unless they take the sample mean out of the original signals beforehand. Therefore, it is necessary for practical signals to give a complete consideration of four possible cases, including known mean, unknown mean, known variance, and unknown variance; moreover, each unknown mean also considers two approaches for comparison: the sample mean and the mean estimated by MLE. In terms of the practical situation of a realization of physical signals, users can choose one case to estimate the Hurst exponent and then the fractal dimension.

The rest of this paper is organized as follows. Section 2 briefly describes mathematical preliminaries. Section 3 introduces practical considerations for the MLE. Section 4 shows how to implement the MLE in an efficient way. Section 5 discusses experimental results. Finally, Section 6 concludes the paper with some facts.

2. Mathematical Preliminaries

In this section, some related models are reviewed, including FBM, FGN, DFBM, and DFGN. For consistency, the notation $\{x(t), t \in \mathbf{R}\}$ is used to denote a continuous-time random process and $\{x[n], n \in \mathbf{Z}\}$ a discrete-time random process.

FBM, represented by $B_H(t, \omega)$ where ω belongs to a sample space Ω , is a generalization of Brownian motion. For conciseness, the short notation $B_H(t)$ is adopted in place of

$B_H(t, \omega)$. According to Mandelbrot and Van Ness [31], FBM is formally defined by the following relations:

$$\begin{aligned} B_H(0) &= b_0, \\ B_H(t) - B_H(0) &= \frac{1}{\Gamma(H + 1/2)} \left\{ \int_{-\infty}^0 [(t-s)^{H-1/2} - (-s)^{H-1/2}] dB(s) \right. \\ &\quad \left. + \int_0^t (t-s)^{H-1/2} dB(s) \right\}, \end{aligned} \quad (1)$$

where H is the Hurst exponent with a value lying between 0 and 1 and the increments of FBM, $dB(t)$, are zero mean, Gaussian, and independent increments of ordinary Brownian motion. Its symmetric form is described as follows:

$$\begin{aligned} B_H(t_2) - B_H(t_1) &= \frac{1}{\Gamma(H + 1/2)} \left\{ \int_0^{t_2} (t_2 - s)^{H-1/2} dB(s) \right. \\ &\quad \left. - \int_0^{t_1} (t_1 - s)^{H-1/2} dB(s) \right\}. \end{aligned} \quad (2)$$

When H equals 0.5, FBM becomes the ordinary Brownian motion. Unfortunately, FBM is a nonstationary process, whose Wigner-Ville spectrum (WVS) is given by the following expression [30]:

$$S_{B_H}(\Omega, t) = (1 - 2^{1-2H} \cos 2\Omega t) \frac{1}{|\Omega|^{2H+1}}. \quad (3)$$

In spite of FBM being a time-varying process, the increment of FBM is a stationary and self-similar process, called FGN.

In real applications, discrete data are collected; sampled data of FBM are expressed as $B_H[n] = B_H(nT_s)$, where T_s is the sampling time. The increments of DFBM, called DFGN, are denoted by $X_H[n] = B_H[n] - B_H[n-1]$. DFGN is a normally distributed and stationary process with zero mean, whose autocorrelation functions (ACFs) are given by the following equation:

$$\begin{aligned} r_H[k] &= E \{x_H[n+k] x_H[n]\} \\ &= \frac{\sigma^2}{2} (|k+1|^{2H} - 2|k|^{2H} + |k-1|^{2H}), \end{aligned} \quad (4)$$

where $\sigma^2 = \text{var}(X_H[k])$ [29, 40]. The ACF, $r_H[k]$, behaves asymptotically as $k^{2H-2} = k^{-\alpha}$, $\alpha \in (0, 2)$ [40].

3. Practical Considerations for the MLE

It is well known from the properties of DFGN that the probability density function (PDF) of DFGN can be expressed as follows [29]:

$$p(\mathbf{x}; H) = \frac{1}{(2\pi)^{N/2} |\mathbf{R}|^{1/2}} \exp \left\{ -\frac{1}{2} \mathbf{x}^T \mathbf{R}^{-1} \mathbf{x} \right\}, \quad (5)$$

where $\mathbf{x} = [x_0 \ x_1 \ \dots \ x_{N-1}]^T$ is the dataset and \mathbf{R} is the autocovariance matrix; that is, $\mathbf{R} = E[\mathbf{x}\mathbf{x}^T]$ or $[\mathbf{R}]_{ij} = r_H(|i - j|)$, where $r_H(k)$ is the ACF as expressed by (4).

In real applications, some physical signals can be modeled as either DFBM or DFGN. If the signals of interest are modeled as DFBM, their increment, DFGN, will not be affected by displacement. However, if the signals of interest are modeled as DFGN, signal displacement will result in a very severe error unless the displacement problem of signals is considered in advance. The reason for displacement may be modeling error, measurement error, inappropriate operation, or apparatus baseline calibrating error, and so forth. In order to avoid the error resulting from displacement, two approaches are considered to estimate displacement: one is to maximize PDF over the mean; the other is to simply take the sample mean out of signals. Considering that the PDF of DFGN has two explicit parameters, the mean and variance, each parameter may be known or unknown, and each unknown mean includes two approaches, all together, there are four cases covering six approaches.

3.1. Case 1: Known Mean (Displacement) and Known Variance.

Under this case, there are no mean and variance necessary to be estimated before estimating the Hurst exponent. In theory, this is the best case since information about the mean and variance is provided. For convenience, the logarithm of PDF will be maximized instead of PDF, which produces the same result since logarithmic operation still preserves the monotonic property of a function. Without loss of generality, displacement is set to be 0. From (5), the logarithm of PDF is as follows:

$$\log p(\mathbf{x}; H) = -\frac{N}{2} \log(2\pi) - \frac{1}{2} \log |\mathbf{R}| - \frac{1}{2} \mathbf{x}^T \mathbf{R}^{-1} \mathbf{x}. \quad (6)$$

Since constant terms and coefficients do not affect the maximum, a compact form is described as follows:

$$\begin{aligned} \max_H \{\log p(\mathbf{x}; H)\} &= \max_H \left\{ -\log |\mathbf{R}| - \mathbf{x}^T \mathbf{R}^{-1} \mathbf{x} \right\} \\ &= \max_H \left\{ -\log |\overline{\mathbf{R}}| - \mathbf{x}^T \overline{\mathbf{R}}^{-1} \mathbf{x} \right\}, \end{aligned} \quad (7)$$

where

$$\mathbf{R} = \sigma^2 \overline{\mathbf{R}}, \quad (8)$$

and σ^2 is known to users.

3.2. Case 2: Known Mean (Displacement) and Unknown Variance. The case first proposed by Lundahl et al. [29] is like this. Likewise, displacement is assumed to be 0. The Hurst exponent can be estimated by using the following equation:

$$\max_H \left[\max_{\sigma^2} \{\log p(\mathbf{x}; H, \sigma^2)\} \right]. \quad (9)$$

It is well known that the logarithm of PDF is expressed as follows:

$$\begin{aligned} \log p(\mathbf{x}; H, \sigma^2) &= -\frac{N}{2} \log(2\pi) - \frac{N}{2} \log \sigma^2 \\ &\quad - \frac{1}{2} \log |\overline{\mathbf{R}}| - \frac{1}{2\sigma^2} \mathbf{x}^T \overline{\mathbf{R}}^{-1} \mathbf{x}. \end{aligned} \quad (10)$$

By maximizing the $\log p(\mathbf{x}; H)$ over σ^2 , it follows that

$$\hat{\sigma}^2 = \frac{1}{N} \mathbf{x}^T \overline{\mathbf{R}}^{-1} \mathbf{x}. \quad (11)$$

By substituting (11) into (10), the final function to be maximized is

$$\begin{aligned} \max_{\sigma^2} \{\log p(\mathbf{x}; H, \sigma^2)\} \\ &= \log p(\mathbf{x}; H, \hat{\sigma}^2) \\ &= -\frac{N}{2} \log(2\pi) - \frac{N}{2} \log(\hat{\sigma}^2) - \frac{1}{2} \log |\overline{\mathbf{R}}| - \frac{N}{2}. \end{aligned} \quad (12)$$

Likewise, the terms that do not affect maximization will be omitted, and thus a compact form is described as follows:

$$\begin{aligned} \max_H \left[\max_{\sigma^2} \{\log p(\mathbf{x}; H, \sigma^2)\} \right] \\ &= \max_H \left[-\log |\overline{\mathbf{R}}| - N \log \left(\frac{1}{N} \mathbf{x}^T \overline{\mathbf{R}}^{-1} \mathbf{x} \right) \right]. \end{aligned} \quad (13)$$

3.3. Case 3: Unknown Mean (Displacement) and Known Variance. Let measurement data be $\mathbf{z} = \mathbf{x} + \boldsymbol{\mu}$, where \mathbf{x} can be modeled as DFGN with zero mean and $\boldsymbol{\mu}$ is a column vector with each element being constant μ ; that is, $\boldsymbol{\mu} = [\mu \ \mu \ \dots \ \mu]^T$. The Hurst exponent can be estimated by using the following two approaches based on two estimators about μ .

Approach 1. First maximize the logarithm of PDF over μ by taking derivative with respect to μ , and then maximize the logarithm of the maximum PDF on estimated μ over H , that is,

$$\max_H \left[\max_{\mu} \{\log p(\mathbf{z}; H, \mu)\} \right]. \quad (14)$$

The unknown displacement of DFGN is assumed to be μ , and thus the PDF will be

$$\begin{aligned} \log p(\mathbf{z}; H, \mu) \\ &= -\frac{N}{2} \log(2\pi) - \frac{1}{2} \log |\mathbf{R}_z| - \frac{1}{2} (\mathbf{z} - \boldsymbol{\mu})^T \mathbf{R}_z^{-1} (\mathbf{z} - \boldsymbol{\mu}), \end{aligned} \quad (15)$$

where $\mathbf{R}_z = E[(\mathbf{z} - \boldsymbol{\mu})(\mathbf{z} - \boldsymbol{\mu})^T] = E[\mathbf{x}\mathbf{x}^T] = \mathbf{R}$. Therefore, (15) can be simplified as

$$\begin{aligned} \log p(\mathbf{z}; H, \mu) \\ &= -\frac{N}{2} \log(2\pi) - \frac{1}{2} \log |\mathbf{R}| - \frac{1}{2} (\mathbf{z} - \boldsymbol{\mu})^T \mathbf{R}^{-1} (\mathbf{z} - \boldsymbol{\mu}). \end{aligned} \quad (16)$$

First, maximize the log $p(\mathbf{z}; H, \mu)$ over μ by taking derivative with respect to μ and the operation is equivalent to maximizing the $(\mathbf{z} - \boldsymbol{\mu})^T \mathbf{R}^{-1} (\mathbf{z} - \boldsymbol{\mu})$. The estimator of μ is derived from the Appendix as follows:

$$\hat{\boldsymbol{\mu}} = \frac{1}{\|\mathbf{A}\|_s} \sum_{k=0}^{N-1} \|\mathbf{a}_k\|_s z_k, \quad (17)$$

where $\mathbf{A} = \mathbf{R}^{-1}$, $\mathbf{a}_k = [a_{0k} \ a_{1k} \ \cdots \ a_{(N-1)k}]^T$, $\|\mathbf{a}_k\|_s = \sum_{i=0}^{N-1} a_{ik}$ and $\|\mathbf{A}\|_s = \sum_{k=0}^{N-1} \|\mathbf{a}_k\|_s$. It is easy to check that

$$\hat{\boldsymbol{\mu}} = \frac{1}{\|\bar{\mathbf{A}}\|_s} \sum_{k=0}^{N-1} \|\bar{\mathbf{a}}_k\|_s z_k, \quad (18)$$

where $\bar{\mathbf{A}} = \bar{\mathbf{R}}^{-1}$, $\bar{\mathbf{a}}_k = [\bar{a}_{0k} \ \bar{a}_{1k} \ \cdots \ \bar{a}_{(N-1)k}]^T$, $\|\bar{\mathbf{a}}_k\|_s = \sum_{i=0}^{N-1} \bar{a}_{ik}$ and $\|\bar{\mathbf{A}}\|_s = \sum_{k=0}^{N-1} \|\bar{\mathbf{a}}_k\|_s$. Next, by substituting (17) into (16), the final function to be maximized is

$$\begin{aligned} & \max_{\mu} \{\log p(\mathbf{z}; H, \mu)\} \\ & = \log p(\mathbf{z}; H, \hat{\boldsymbol{\mu}}) \\ & = -\frac{N}{2} \log(2\pi) - \frac{1}{2} \log |\mathbf{R}| - \frac{1}{2} (\mathbf{z} - \hat{\boldsymbol{\mu}})^T \mathbf{R}^{-1} (\mathbf{z} - \hat{\boldsymbol{\mu}}). \end{aligned} \quad (19)$$

Likewise, the terms without affecting maximization are omitted, and thus a compact form is described as follows:

$$\begin{aligned} & \max_H \left[\max_{\mu} \{\log p(\mathbf{z}; H, \mu)\} \right] \\ & = \max_H \{\log p(\mathbf{z}; H, \hat{\boldsymbol{\mu}})\} \\ & = \max_H \left[-\log |\bar{\mathbf{R}}| - (\mathbf{z} - \hat{\boldsymbol{\mu}})^T \bar{\mathbf{R}}^{-1} (\mathbf{z} - \hat{\boldsymbol{\mu}}) \right]. \end{aligned} \quad (20)$$

Approach 2. Use the sample mean to replace the previous estimator of μ . Other procedures are the same as the ones of Approach 1. The sample mean is the simplest method to estimate the mean, which is

$$\hat{\boldsymbol{\mu}} = \frac{1}{N} \sum_{k=0}^{N-1} z_k. \quad (21)$$

3.4. Case 4: Unknown Mean (Displacement) and Unknown Variance. This case is the most general in real applications. Like Case 3, measurement data are assumed to be $\mathbf{z} = \mathbf{x} + \boldsymbol{\mu}$ and the unknown variance is σ^2 . Similarly, the Hurst exponent is estimated by using the following two approaches.

Approach 1. Like Case 3, how to estimate the Hurst exponent is described as follows:

$$\max_H \left[\max_{\sigma^2, \mu} \{\log p(\mathbf{z}; H, \sigma^2, \mu)\} \right]. \quad (22)$$

First, maximize the log $p(\mathbf{x}; H, \mu)$ over σ^2 and μ by taking derivatives with respect to σ^2 and μ , respectively, and then the estimators of σ^2 and μ are derived as follows:

$$\hat{\sigma}^2 = \frac{1}{N} (\mathbf{z} - \hat{\boldsymbol{\mu}})^T \bar{\mathbf{R}}^{-1} (\mathbf{z} - \hat{\boldsymbol{\mu}}), \quad \hat{\boldsymbol{\mu}} = \frac{1}{\|\bar{\mathbf{A}}\|_s} \sum_{k=0}^{N-1} \|\bar{\mathbf{a}}_k\|_s z_k. \quad (23)$$

Likewise, the terms that do not affect maximization are omitted, and thus a compact form is described as follows:

$$\begin{aligned} & \max_H \left[\max_{\sigma^2, \mu} \{\log p(\mathbf{z}; H, \sigma^2, \mu)\} \right] \\ & = \max_H \left[-\log |\bar{\mathbf{R}}| - N \log \left(\frac{1}{N} (\mathbf{z} - \hat{\boldsymbol{\mu}})^T \bar{\mathbf{R}}^{-1} (\mathbf{z} - \hat{\boldsymbol{\mu}}) \right) \right]. \end{aligned} \quad (24)$$

Approach 2. Use the sample mean to replace the previous estimator of μ . Other procedures are the same as the ones of Approach 1.

The final step of each case is to estimate the Hurst exponent, but it needs some tips. A direct maximization over H is infeasible because the Hurst exponent is an implicit parameter. Therefore, the golden section search [41] was adopted to find out the maxima of (7), (13), (20), and (24) in this paper.

4. Efficient Procedures for the MLE

In this section, the computational stability and efficiency of using the MLE for the Hurst exponent are studied. Since computing the inverse and determinant of an autocovariance matrix is sensitive to the data size, especially when H is close to 1 [35]. Also, for a large dataset, storing the whole autocovariance matrix requires a large amount of computer memory. Therefore, a reliable and efficient procedure is necessary for estimating the Hurst exponent, especially when users use an ordinary computer with less memory and lower CPU speed. After carefully studying the structure of an autocovariance matrix, a combination of the Levinson algorithm and Cholesky decomposition can be applied to efficiently compute the inverse and determinant of an autocovariance matrix, and then the iterative structures of the two algorithms can be exploited to estimate the Hurst exponent without storing any matrix and performing any matrix multiplication. For convenience, some notations are listed below for further quotation:

$$\begin{aligned} \bar{\mathbf{R}}^{-1} &= \mathbf{L}^T \mathbf{D} \mathbf{L} = (\mathbf{D}^{1/2} \mathbf{L})^T (\mathbf{D}^{1/2} \mathbf{L}) \equiv \mathbf{W}^T \mathbf{W}, \\ \mathbf{D} &= \text{diag}(P_0^{-1}, P_1^{-1}, \dots, P_{N-1}^{-1}), \end{aligned}$$

$$\mathbf{L} = \begin{bmatrix} 1 & 0 & \cdots & 0 \\ a_1(1) & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ a_{N-1}(N-1) & a_{N-1}(N-2) & \cdots & 1 \end{bmatrix} = \begin{bmatrix} \mathbf{a}_0^T \\ \mathbf{a}_1^T \\ \vdots \\ \mathbf{a}_{N-1}^T \end{bmatrix},$$

$$\mathbf{a}_i^T = [a_i(i) \ \cdots \ a_i(1) \ 1 \ 0 \ \cdots \ 0],$$

$$i = 0, 1, \dots, N-1,$$

$$\mathbf{W} = \begin{bmatrix} P_0^{1/2} & 0 & \cdots & 0 \\ P_1^{1/2} a_1(1) & P_1^{1/2} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ P_{N-1}^{1/2} a_{N-1}(N-1) & P_{N-1}^{1/2} a_{N-1}(N-2) & \cdots & P_{N-1}^{1/2} \end{bmatrix}$$

$$= \begin{bmatrix} \mathbf{w}_0^T \\ \mathbf{w}_1^T \\ \vdots \\ \mathbf{w}_{N-1}^T \end{bmatrix} \quad (25)$$

$$\mathbf{W}_i^T = P_i^{1/2} \mathbf{a}_i^T, \quad i = 0, 1, \dots, N-1, \quad (26)$$

$$|\bar{\mathbf{R}}| = \prod_{i=0}^{N-1} P_i, \quad (27)$$

where $a_M(k), k = 1, 2, \dots, M$ are the predictor coefficients of order M and $P_i, i = 0, 1, \dots, N-1$ are the prediction error powers of order $N-1$. These coefficients are iteratively computed by the Levinson algorithm. It is worth noting that when numerically calculating $\log |\bar{\mathbf{R}}|$, $\sum_{i=0}^{N-1} \log P_i$ is computed instead of $\log(P_0 P_1 \cdots P_{N-1})$ because the term $P_0 P_1 \cdots P_{N-1}$ easily approaches to zero numerically as the data size grows larger. Thus, using the following equation to compute $\log |\bar{\mathbf{R}}|$ is essential:

$$\log |\bar{\mathbf{R}}| = \sum_{i=0}^{N-1} \log P_i. \quad (28)$$

Obviously, the Levinson algorithm and Cholesky decomposition can be used to save the time of computing the inverse and determinant of the autocovariance matrix. However, estimating the Hurst exponent by the currently mentioned structure of implementation still needs matrix computation and storing, which requires excessive computational time and computer memory. When taking a closer look at the term $\mathbf{x}^T \bar{\mathbf{R}}^{-1} \mathbf{x}$ of (7) or (13), a magic and helpful form appears as follows:

$$\mathbf{x}^T \bar{\mathbf{R}}^{-1} \mathbf{x} = (\mathbf{W}\mathbf{x})^T (\mathbf{W}\mathbf{x}) \equiv \mathbf{y}^T \mathbf{y}, \quad (29)$$

where

$$\mathbf{y} \equiv [y_0 \ y_1 \ \cdots \ y_{N-1}]^T, \quad (30)$$

$$y_i = \mathbf{w}_i^T \mathbf{x}, \quad i = 0, 1, \dots, N-1. \quad (31)$$

With (31), storing any matrix in the process of computation is no longer necessary, which is also a very efficient step.

In this paper, the golden section search was adopted to find out the maxima of (7), (13), (20), and (24). In the process of searching for each maximum, computing the inner terms of (7), (13), (20), or (24) is necessary, such as $-\log |\bar{\mathbf{R}}| - \mathbf{x}^T \bar{\mathbf{R}}^{-1} \mathbf{x}$, $-\log |\bar{\mathbf{R}}| - N \log(\mathbf{x}^T \bar{\mathbf{R}}^{-1} \mathbf{x}/N)$, $-\log |\bar{\mathbf{R}}| - (\mathbf{z} - \hat{\boldsymbol{\mu}})^T \bar{\mathbf{R}}^{-1} (\mathbf{z} - \hat{\boldsymbol{\mu}})$ or $-\log |\bar{\mathbf{R}}| - N \log((\mathbf{z} - \hat{\boldsymbol{\mu}})^T \bar{\mathbf{R}}^{-1} (\mathbf{z} - \hat{\boldsymbol{\mu}})/N)$.

In order to efficiently estimate the Hurst exponent by using (7) or (13), first compute $\mathbf{a}_i, i = 0, 1, \dots, N-1$, by using the Levinson algorithm, then \mathbf{w}_i by using (26), \mathbf{y} by using (30) and (31), $\mathbf{x}^T \bar{\mathbf{R}}^{-1} \mathbf{x}$ by using (29), and $\log |\bar{\mathbf{R}}|$ by using (28). The details of determining the Hurst exponent by using the MLE are described in the following procedure.

Procedure 1. By efficiently computing $-\log |\bar{\mathbf{R}}| - \mathbf{x}^T \bar{\mathbf{R}}^{-1} \mathbf{x}$ or $-\log |\bar{\mathbf{R}}| - N \log(\mathbf{x}^T \bar{\mathbf{R}}^{-1} \mathbf{x}/N)$, consider the following steps:

- (1) initialize $i = 0$;
- (2) compute \mathbf{a}_i and P_i by using the Levinson algorithm;
- (3) compute \mathbf{w}_i^T by using (26);
- (4) compute $y_i = \mathbf{w}_i^T \mathbf{x}$;
- (5) add a new element into the vector \mathbf{y} like (30);
- (6) perform $i = i + 1$; if $i \leq N-1$, then go to Step 2 or go to the next step;
- (7) compute $\mathbf{x}^T \bar{\mathbf{R}}^{-1} \mathbf{x}$ using (29) and $\log |\bar{\mathbf{R}}|$ using (28);
- (8) compute $-\log |\bar{\mathbf{R}}| - \mathbf{x}^T \bar{\mathbf{R}}^{-1} \mathbf{x}$ for (7) or $-\log |\bar{\mathbf{R}}| - N \log(\mathbf{x}^T \bar{\mathbf{R}}^{-1} \mathbf{x}/N)$ for (13).

Obviously, it is unnecessary to store any matrix and execute any matrix multiplication in a series of computation except for vector storing and multiplication. The efficient procedure not only saves computer memory but also storing time.

Next, an efficient procedure for computing (20) or (24) is considered. Each procedure considers two approaches to estimating the mean: the sample mean and the mean estimated by MLE. For the first approach, users first take the sample mean out of the original signals and then call the function evaluation of (7) or (13). For the second approach, users must use an efficient implementing procedure for (17) to estimate the mean. Based on the composition structure of the mean estimated by MLE, (17) can be decomposed into the following equation:

$$\hat{\boldsymbol{\mu}} = \frac{\mathbf{1}^T \mathbf{A} \mathbf{z}}{\mathbf{1}^T \mathbf{A} \mathbf{1}} = \frac{\mathbf{1}^T \mathbf{W}^T \mathbf{W} \mathbf{z}}{\mathbf{1}^T \mathbf{W}^T \mathbf{W} \mathbf{1}} = \frac{(\mathbf{W} \mathbf{1})^T (\mathbf{W} \mathbf{z})}{(\mathbf{W} \mathbf{1})^T (\mathbf{W} \mathbf{1})}, \quad (32)$$

where

$$\mathbf{1} = [1 \ 1 \ \cdots \ 1]^T, \quad (33)$$

$$\mathbf{W} \mathbf{1} = [\mathbf{w}_0^T \mathbf{1} \ \mathbf{w}_1^T \mathbf{1} \ \cdots \ \mathbf{w}_{N-1}^T \mathbf{1}]^T. \quad (34)$$

When carefully observing the term $(\mathbf{z} - \hat{\boldsymbol{\mu}})^T \bar{\mathbf{R}}^{-1} (\mathbf{z} - \hat{\boldsymbol{\mu}})$, it can be decomposed into the following equation:

$$(\mathbf{z} - \hat{\boldsymbol{\mu}})^T \bar{\mathbf{R}}^{-1} (\mathbf{z} - \hat{\boldsymbol{\mu}}) = (\mathbf{W}\mathbf{z} - \mathbf{W}\hat{\boldsymbol{\mu}})^T (\mathbf{W}\mathbf{z} - \mathbf{W}\hat{\boldsymbol{\mu}}), \quad (35)$$

where

$$\mathbf{W}\mathbf{z} = [\mathbf{w}_0^T \mathbf{z} \quad \mathbf{w}_1^T \mathbf{z} \quad \cdots \quad \mathbf{w}_{N-1}^T \mathbf{z}]^T, \quad (36)$$

$$\mathbf{W}\hat{\boldsymbol{\mu}} = \hat{\boldsymbol{\mu}} \mathbf{W}\mathbf{1}. \quad (37)$$

In order to efficiently compute (20) or (24), first, compute $\mathbf{a}_i, i = 0, 1, \dots, N-1$, by using the Levinson algorithm, then, \mathbf{w}_i by using (26), $\mathbf{W}\mathbf{1}$ by using (34), $\mathbf{W}\mathbf{z}$ by using (36), $\hat{\boldsymbol{\mu}}$ by using (32), $\mathbf{W}\hat{\boldsymbol{\mu}}$ by using (37), $(\mathbf{z} - \hat{\boldsymbol{\mu}})^T \bar{\mathbf{R}}^{-1} (\mathbf{z} - \hat{\boldsymbol{\mu}})$ by using (35), and $\log |\bar{\mathbf{R}}|$ by using (28). The details of determining the Hurst exponent by using the MLE are described in the following procedure.

Procedure 2. By efficiently computing $-\log |\bar{\mathbf{R}}| - (\mathbf{z} - \hat{\boldsymbol{\mu}})^T \bar{\mathbf{R}}^{-1} (\mathbf{z} - \hat{\boldsymbol{\mu}})$ or $-\log |\bar{\mathbf{R}}| - N \log((\mathbf{z} - \hat{\boldsymbol{\mu}})^T \bar{\mathbf{R}}^{-1} (\mathbf{z} - \hat{\boldsymbol{\mu}})/N)$, consider the following steps:

- (1) initialize $i = 0$;
- (2) compute \mathbf{a}_i and P_i using the Levinson algorithm;
- (3) compute \mathbf{w}_i^T by using (26);
- (4) compute $\mathbf{w}_i^T \mathbf{1}$ and $\mathbf{w}_i^T \mathbf{z}$;
- (5) add a new element into the vector $\mathbf{W}\mathbf{1}$ like (34) and $\mathbf{W}\mathbf{z}$ like (36);
- (6) perform $i = i + 1$; if $i \leq N - 1$, then go to Step 2 or go to the next step;
- (7) compute $\hat{\boldsymbol{\mu}}$ by using (32) and $\mathbf{W}\hat{\boldsymbol{\mu}}$ by using (37);
- (8) compute $(\mathbf{z} - \hat{\boldsymbol{\mu}})^T \bar{\mathbf{R}}^{-1} (\mathbf{z} - \hat{\boldsymbol{\mu}})$ by using (35) and $\log |\bar{\mathbf{R}}|$ by using (28);
- (9) compute $-\log |\bar{\mathbf{R}}| - (\mathbf{z} - \hat{\boldsymbol{\mu}})^T \bar{\mathbf{R}}^{-1} (\mathbf{z} - \hat{\boldsymbol{\mu}})$ for (20) or $-\log |\bar{\mathbf{R}}| - N \log((\mathbf{z} - \hat{\boldsymbol{\mu}})^T \bar{\mathbf{R}}^{-1} (\mathbf{z} - \hat{\boldsymbol{\mu}})/N)$ for (24).

Similar to Procedure 1, it is not necessary to store any matrix and execute matrix multiplication in a series of computation except for vector storing and multiplication. Without considering the efficient computation of $\mathbf{x}^T \bar{\mathbf{R}}^{-1} \mathbf{x}$, users first implement the Levinson algorithm to obtain \mathbf{a}_i and $P_i, i = 0, 1, \dots, N-1$ and then store \mathbf{L} and \mathbf{D} , obtain \mathbf{R}^{-1} by using $\mathbf{L}^T \mathbf{D} \mathbf{L}$, and finally take matrix computation for $\mathbf{x}^T \bar{\mathbf{R}}^{-1} \mathbf{x}$. The traditional procedure only saves the time of inverting matrix by using the Levinson algorithm but overlooks the potential efficacy of the predictor coefficients and prediction error powers iteratively generated. With this stable and efficient implementation, the practicability of the MLE will be greatly enhanced.

5. Results and Discussion

In order to analyze four possible cases and compare their efficiency, the generating algorithm proposed by Lundahl et

al. [29] was used to generate DFGN because the realizations produced by this algorithm possess fine correlation structure and long-term dependency.

For more convincing facts, a wider range of Hurst exponents and data sizes were considered, including $H = 0.01, 0.05, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 0.95$, and 0.99 (totally, 13 Hurst exponents), as well as $N = 128, 256, 512, 1024, 2048$, and 4096 (totally, 6 types of data sizes). For each data size, 100 realizations of white Gaussian noise were generated by a Gaussian random generator to form 100 realizations of DFGN for each Hurst exponent.

All estimations were performed with the same computing specifications: (1) hardware: a computer of Intel Core i7-2600 processor up to 3.40 GHz and a RAM of 8.00 GB (7.89 GB available); (2) operating system: Windows 7 Professional Service Pack 1; (3) programming software: MATLAB R2011b 64-bit (win64); (4) optimization algorithm: golden section search with threshold 0.0001, which takes 21 iterations and totally 22 function evaluations [42]. Table 1 shows the experimental results, each value representing the mean of mean-squared errors (MSEs) of 100 realizations over 13 Hurst exponents, simply denoted as mean mean-squared errors (MMSEs).

On the other hand, the function evaluation time is recorded and is used to compare with the implementation time of the traditional MLE for efficiency analysis. Table 2 lists the average time (in seconds) of 13 Hurst exponents spent by each approach in two executing procedures, with and without considering the computational efficiency, as well as their corresponding time ratio.

The best results are almost acquired among Case 1, and the second best results are among Case 3, and the worst results are among Case 4 from Table 1. This is reasonable because both mean and variance are known for Case 1, but both mean and variance are unknown for Case 4. It is worth noting that the accuracy of Case 3 is better than Case 2. This indicates that accuracy is more related to known variance than known mean. Generally speaking, a most practical situation is Case 4 with both unknown mean and variance; the situation of Case 1 is less likely practical. Table 1 suggests that using the sample mean instead of the mean estimated by MLE is also a reliable approach.

It is easy to see that the computational cost of the traditional MLE for the Hurst exponent needs $O(N^3)$, whereas two newly proposed procedures only need $O(N^2)$. In addition to lower computational complexity, storing data by vector instead of matrix also help raise the computational efficacy. Table 2 suggests that without matrix calculation, time saving is obvious, especially when the data size grows larger. For example, with the data size of 4096, the ratio of each proposed efficient procedure to the traditional one reaches up to 80 times. The ratio will be more tremendous especially for computers of limited resources. These results will contribute to the position of the MLE for estimating Hurst exponent.

6. Conclusions

In parameter estimation, both accuracy and efficiency are generally difficult to coexist. Accordingly, how to weigh

TABLE 1: Accuracy comparison for four cases covering six approaches, each value representing the mean of mean-squared errors (MSEs) of 100 realizations over 13 Hurst exponents, simply denoted as mean mean-squared errors (MMSEs).

Cases	$N = 128$	$N = 256$	$N = 512$	$N = 1024$	$N = 2048$	$N = 4096$
C1	1.18E - 03	5.52E - 04	2.90E - 04	1.51E - 04	7.11E - 05	3.73E - 05
C2	1.72E - 03	8.50E - 04	4.65E - 04	2.59E - 04	1.19E - 04	6.57E - 05
C3-A1 ^a	1.36E - 03	5.99E - 04	2.94E - 04	1.55E - 04	7.44E - 05	3.71E - 05
C3-A2 ^a	1.35E - 03	6.04E - 04	2.94E - 04	1.57E - 04	7.45E - 05	3.74E - 05
C4-A1 ^a	2.51E - 03	1.15E - 03	5.18E - 04	2.97E - 04	1.38E - 04	6.76E - 05
C4-A2 ^a	2.49E - 03	1.15E - 03	5.19E - 04	2.98E - 04	1.38E - 04	6.81E - 05

^aA1 denotes Approach 1 and A2 denotes Approach 2.

TABLE 2: Efficiency comparison for four cases covering six approaches, each value representing either the average time (in seconds) of 13 Hurst exponents spent by each approach in two executing procedures (with and without considering the computational efficiency) or their corresponding time ratio.

Cases	$N = 128$	$N = 256$	$N = 512$	$N = 1024$	$N = 2048$	$N = 4096$
C1	1.27E - 01	3.98E - 01	1.64E + 00	1.06E + 01	1.11E + 02	8.78E + 02
C1 ^b	1.02E - 01	2.03E - 01	4.40E - 01	1.05E + 00	2.88E + 00	1.04E + 01
Ratio	1.25E + 00	1.97E + 00	3.73E + 00	1.01E + 01	3.86E + 01	8.46E + 01
C2	1.19E - 01	3.63E - 01	1.61E + 00	1.03E + 01	1.11E + 02	8.70E + 02
C2 ^b	9.83E - 02	2.02E - 01	4.37E - 01	1.16E + 00	2.88E + 00	1.03E + 01
Ratio	1.21E + 00	1.80E + 00	3.68E + 00	8.94E + 00	3.85E + 01	8.47E + 01
C3-A1 ^a	1.29E - 01	3.62E - 01	1.65E + 00	1.05E + 01	1.09E + 02	8.80E + 02
C3-A1 ^{a,b}	1.09E - 01	2.14E - 01	4.57E - 01	1.09E + 00	2.93E + 00	1.09E + 01
Ratio	1.18E + 00	1.69E + 00	3.60E + 00	9.71E + 00	3.73E + 01	8.08E + 01
C3-A2 ^a	1.27E - 01	3.50E - 01	1.59E + 00	1.06E + 01	1.10E + 02	8.70E + 02
C3-A2 ^{a,b}	1.00E - 01	2.05E - 01	4.36E - 01	1.05E + 00	2.88E + 00	1.03E + 01
Ratio	1.27E + 00	1.71E + 00	3.64E + 00	1.01E + 01	3.82E + 01	8.42E + 01
C4-A1 ^a	1.16E - 01	3.74E - 01	1.68E + 00	1.06E + 01	1.10E + 02	8.71E + 02
C4-A1 ^{a,b}	1.01E - 01	2.14E - 01	4.52E - 01	1.08E + 00	2.93E + 00	1.06E + 01
Ratio	1.15E + 00	1.74E + 00	3.71E + 00	9.78E + 00	3.76E + 01	8.18E + 01
C4-A2 ^a	1.17E - 01	3.53E - 01	1.65E + 00	1.03E + 01	1.10E + 02	8.78E + 02
C4-A2 ^{a,b}	9.44E - 02	2.02E - 01	4.42E - 01	1.04E + 00	2.88E + 00	1.07E + 01
Ratio	1.24E + 00	1.75E + 00	3.75E + 00	9.88E + 00	3.83E + 01	8.21E + 01

^aA1 denotes Approach 1 and A2 denotes Approach 2

^bEstimation implemented by the two newly proposed procedures.

the accuracy and efficiency before estimating parameters is usually a matter of a dilemma. The MLE for Hurst exponent is considered optimal in accuracy, whereas the computational cost of the MLE was once considered as tremendous, which hinders the MLE from being recommended to quick response systems. Fortunately, the Levinson algorithm and Cholesky decomposition can be combined to improve the computational efficiency, and further overcome the dilemma. On the other hand, a potential modeling problem of physical signals is also considered. The first proposed MLE for Hurst exponent only considered one case with given mean as zero, which is only suitable for signals of DFBM. However, many physical signals are like the model of DFGN, with nonzero means. Therefore, users must take the sample mean out of the original signal before using the MLE, or a direct computation will easily lead to a severely wrong result, further providing a wrong signal explanation. In order to extend the MLE for Hurst exponent to signals of DFGN, four possible cases are considered: known mean, unknown mean, known variance,

and unknown variance. The experimental results show that the computational cost is largely reduced by a combination of Levinson algorithm and Cholesky decomposition. Moreover, numerical stability is also provided to help users avoid numerical mistakes due to negligence. After balancing inherent accuracy with boosted efficiency, the MLE might be the preferred option for estimating Hurst exponent in the near future. More importantly, this idea for efficiently implementing the MLE can be extended to other variants of the MLE for other fields, making real-time computation with best accuracy more possible.

Appendix

Proof of Case 3. In this appendix, maximizing $(\mathbf{z} - \boldsymbol{\mu})^T \mathbf{A} (\mathbf{z} - \boldsymbol{\mu})$ with respect to $\boldsymbol{\mu}$ is proved to be $\hat{\boldsymbol{\mu}} = (1/\|\mathbf{A}\|_s) \sum_{k=0}^{N-1} \|\mathbf{a}_k\|_s z_k$, where $\mathbf{A} = \mathbf{R}^{-1}$, $\mathbf{a}_k = [a_{0k} \ a_{1k} \ \cdots \ a_{(N-1)k}]^T$, $\|\mathbf{a}_k\|_s = \sum_{i=0}^{N-1} a_{ik}$, and $\|\mathbf{A}\|_s = \sum_{k=0}^{N-1} \|\mathbf{a}_k\|_s$ by using

mathematical induction. Obviously, $\|\mathbf{A}\|_s$ denotes the sum of all elements of matrix \mathbf{A} . For clarity, the subscript N is used to emphasize the dependence on the data size during the procedure of proof. Under this situation, $\hat{\mu} = (1/\|\mathbf{A}_N\|_s) \sum_{k=0}^{N-1} \|\mathbf{a}_{k,N}\|_s z_k$, where $\mathbf{A}_N = \mathbf{R}_N^{-1}$, $\mathbf{a}_{k,N} = [a_{0k} \ a_{1k} \ \cdots \ a_{(N-1)k}]^T$, $\|\mathbf{a}_{k,N}\|_s = \sum_{i=0}^{N-1} a_{ik}$ and $\|\mathbf{A}_N\|_s = \sum_{k=0}^{N-1} \|\mathbf{a}_{k,N}\|_s$.

For $N = 1$, the trivial case, it follows that

$$\begin{aligned} (\mathbf{z} - \boldsymbol{\mu})_1^T \mathbf{A}_1 (\mathbf{z} - \boldsymbol{\mu})_1 &= (z_0 - \mu) a_{00} (z_0 - \mu) \\ &= a_{00} (z_0^2 - 2z_0\mu + \mu^2). \end{aligned} \quad (\text{A.1})$$

By maximizing the above quantity, it follows that

$$\frac{\partial}{\partial \mu} (\mathbf{z} - \boldsymbol{\mu})^T \mathbf{A}_1 (\mathbf{z} - \boldsymbol{\mu}) = a_{00} (-2z_0 + 2\mu) = 0. \quad (\text{A.2})$$

Therefore, it follows that $\hat{\mu} = (1/a_{00})a_{00}z_0 = z_0$, which is consistent with the equality $\hat{\mu} = (1/\|\mathbf{A}_N\|_s) \sum_{k=0}^{N-1} \|\mathbf{a}_{k,N}\|_s z_k$, for $N = 1$, as desired. So, the proposition is true for $N = 1$. Next, assume that $\hat{\mu} = (1/\|\mathbf{A}_M\|_s) \sum_{k=0}^{M-1} \|\mathbf{a}_{k,M}\|_s z_k$, for some integers $M > 1$; that is,

$$\begin{aligned} \frac{\partial}{\partial \mu} (\mathbf{z} - \boldsymbol{\mu})^T \mathbf{A}_M (\mathbf{z} - \boldsymbol{\mu})_M &= 2\mu \sum_{k=0}^{M-1} \|\mathbf{a}_{k,M}\|_s - 2 \sum_{k=0}^{M-1} \|\mathbf{a}_{k,M}\|_s z_k \\ &= 2\mu \|\mathbf{A}_M\|_s - 2 \sum_{k=0}^{M-1} \|\mathbf{a}_{k,M}\|_s z_k = 0. \end{aligned} \quad (\text{A.3})$$

Finally, let $N = M + 1$; then it follows that

$$\begin{aligned} &(\mathbf{z} - \boldsymbol{\mu})_{M+1}^T \mathbf{A}_{M+1} (\mathbf{z} - \boldsymbol{\mu})_{M+1} \\ &= [(\mathbf{z} - \boldsymbol{\mu})_M^T \ z_M - \mu] \begin{bmatrix} \mathbf{A}_M & \mathbf{a}'_{M,M+1} \\ \mathbf{a}_{M,M+1}^T & a_{MM} \end{bmatrix} \begin{bmatrix} (\mathbf{z} - \boldsymbol{\mu})_M \\ z_M - \mu \end{bmatrix}, \\ &= [(\mathbf{z} - \boldsymbol{\mu})_M^T \ z_M - \mu] \\ &\quad \times \begin{bmatrix} \mathbf{A}_M (\mathbf{z} - \boldsymbol{\mu})_M + \mathbf{a}'_{M,M+1} (z_M - \mu) \\ \mathbf{a}_{M,M+1}^T (\mathbf{z} - \boldsymbol{\mu})_M + a_{MM} (z_M - \mu) \end{bmatrix} \\ &= (\mathbf{z} - \boldsymbol{\mu})_M^T \mathbf{A}_M (\mathbf{z} - \boldsymbol{\mu})_M + (\mathbf{z} - \boldsymbol{\mu})_M^T \mathbf{a}'_{M,M+1} (z_M - \mu) \\ &\quad + \mathbf{a}_{M,M+1}^T (\mathbf{z} - \boldsymbol{\mu})_M (z_M - \mu) + a_{MM} (z_M - \mu)^2 \\ &= (\mathbf{z} - \boldsymbol{\mu})_M^T \mathbf{A}_M (\mathbf{z} - \boldsymbol{\mu})_M \\ &\quad + 2\mathbf{a}_{M,M+1}^T (\mathbf{z} - \boldsymbol{\mu})_M (z_M - \mu) + a_{MM} (z_M - \mu)^2 \\ &= (\mathbf{z} - \boldsymbol{\mu})_M^T \mathbf{A}_M (\mathbf{z} - \boldsymbol{\mu})_M \\ &\quad + 2a_{0M} (z_0 - \mu) (z_M - \mu) + 2a_{1M} (z_1 - \mu) (z_M - \mu) \\ &\quad + \cdots + 2a_{(M-1)M} (z_{M-1} - \mu) (z_M - \mu) + a_{MM} (z_M - \mu)^2 \\ &\quad \mathbf{a}'_{M,M+1} \equiv [a_{0M} \ a_{1M} \ \cdots \ a_{(M-1)M}]^T. \end{aligned} \quad (\text{A.4})$$

By maximizing the quantity with respect to μ , it follows that

$$\begin{aligned} &\frac{\partial}{\partial \mu} (\mathbf{z} - \boldsymbol{\mu})_{M+1}^T \mathbf{A}_{M+1} (\mathbf{z} - \boldsymbol{\mu})_{M+1} \\ &= \frac{\partial}{\partial \mu} (\mathbf{z} - \boldsymbol{\mu})_M^T \mathbf{A}_M (\mathbf{z} - \boldsymbol{\mu})_M \\ &\quad + 2a_{0M} (z_M - \mu) (-1) + 2a_{0M} (z_0 - \mu) (-1) \\ &\quad + 2a_{1M} (z_M - \mu) (-1) + 2a_{1M} (z_1 - \mu) (-1) \\ &\quad + \cdots + 2a_{(M-1)M} (z_M - \mu) (-1) \\ &\quad + 2a_{(M-1)M} (z_{M-1} - \mu) (-1) + 2a_{MM} (z_M - \mu) (-1) \\ &= 2\mu \|\mathbf{A}_M\|_s - 2 \sum_{k=0}^{M-1} \|\mathbf{a}_{k,M}\|_s z_k \\ &\quad + 2\mu (2a_{0M} + 2a_{1M} + \cdots + 2a_{(M-1)M} + a_{MM}) - 2a_{0M} z_0 \\ &\quad - 2a_{1M} z_1 - \cdots - 2a_{(M-1)M} z_{M-1} \\ &\quad - 2(a_{0M} + a_{1M} + \cdots + a_{MM}) z_M \\ &= 2\mu (\|\mathbf{A}_M\|_s + 2a_{0M} + 2a_{1M} + \cdots + 2a_{(M-1)M} + a_{MM}) \\ &\quad - 2 \sum_{k=0}^{M-1} (\|\mathbf{a}_{k,M}\|_s + a_{kM}) z_k \\ &\quad - 2(a_{0M} + a_{1M} + \cdots + a_{MM}) z_M, \\ &= 2\mu \|\mathbf{A}_{M+1}\|_s - 2 \sum_{k=0}^{M-1} \|\mathbf{a}_{k,M+1}\|_s z_k - 2\|\mathbf{a}_{M,M+1}\|_s z_M \\ &= 2\mu \|\mathbf{A}_{M+1}\|_s - 2 \sum_{k=0}^M \|\mathbf{a}_{k,M+1}\|_s z_k = 0, \quad a_{ij} = a_{ji}. \end{aligned} \quad (\text{A.5})$$

Therefore, it follows that

$$\hat{\mu} = \frac{1}{\|\mathbf{A}_{M+1}\|_s} \sum_{k=0}^M \|\mathbf{a}_{k,M+1}\|_s z_k, \quad (\text{A.6})$$

as desired. Kay [43] also provides another derivation from a more general form of a linear model without considering the Hurst exponent H . \square

Conflict of Interests

The author declares that there is no conflict of interests regarding the publication of this paper.

References

- [1] Y.-Z. Wang, B. Li, R.-Q. Wang, J. Su, and X.-X. Rong, "Application of the Hurst exponent in ecology," *Computers & Mathematics with Applications*, vol. 61, no. 8, pp. 2129–2131, 2011.
- [2] B. B. Mandelbrot, *The Fractal Geometry of Nature*, W. H. Freeman, New York, NY, USA, 1983.

- [3] A. P. Pentland, "Fractal-based description of natural scenes," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 6, no. 6, pp. 661–674, 1984.
- [4] C. M. Hagerhall, T. Purcell, and R. Taylor, "Fractal dimension of landscape silhouette outlines as a predictor of landscape preference," *Journal of Environmental Psychology*, vol. 24, no. 2, pp. 247–255, 2004.
- [5] W. N. Gonçalves and O. M. Bruno, "Combining fractal and deterministic walkers for texture analysis and classification," *Pattern Recognition*, vol. 46, no. 11, pp. 2953–2968, 2013.
- [6] A. G. Zuñiga, J. B. Florindo, and O. M. Bruno, "Gabor wavelets combined with volumetric fractal dimension applied to texture analysis," *Pattern Recognition Letters*, vol. 36, pp. 135–143, 2014.
- [7] S. Chang, S.-T. Mao, S.-J. Hu, W.-C. Lin, and C.-L. Cheng, "Studies of detrusor-sphincter synergia and dyssynergia during micturition in rats via fractional Brownian motion," *IEEE Transactions on Biomedical Engineering*, vol. 47, no. 8, pp. 1066–1073, 2000.
- [8] S. Chang, S.-J. Hu, and W.-C. Lin, "Fractal dynamics and synchronization of rhythms in urodynamics of female Wistar rats," *Journal of Neuroscience Methods*, vol. 139, no. 2, pp. 271–279, 2004.
- [9] S. Chang, S.-J. Li, M.-J. Chiang, S.-J. Hu, and M.-C. Hsyu, "Fractal dimension estimation via spectral distribution function and its application to physiological signals," *IEEE Transactions on Biomedical Engineering*, vol. 54, no. 10, pp. 1895–1898, 2007.
- [10] S. Chang, M.-C. Hsyu, H.-Y. Cheng, and S.-H. Hsieh, "Synergic co-activation of muscles in elbow flexion via fractional Brownian motion," *The Chinese Journal of Physiology*, vol. 51, no. 6, pp. 376–386, 2008.
- [11] S. Chang, "Physiological rhythms, dynamical diseases and acupuncture," *The Chinese Journal of Physiology*, vol. 53, no. 2, pp. 77–90, 2010.
- [12] S. Chang, "Fractional Brownian motion in biomedical signal processing, physiology, and modern physics," in *Proceedings of the 5th International Conference on Bioinformatics and Biomedical Engineering (iCBBE '11)*, Wuhan, China, May 2011.
- [13] P.-W. Huang and C.-H. Lee, "Automatic classification for pathological prostate images based on fractal analysis," *IEEE Transactions on Medical Imaging*, vol. 28, no. 7, pp. 1037–1050, 2009.
- [14] P.-L. Lin, P.-W. Huang, C.-H. Lee, and M.-T. Wu, "Automatic classification for solitary pulmonary nodule in CT image by fractal analysis based on fractional Brownian motion model," *Pattern Recognition*, vol. 46, no. 12, pp. 3279–3287, 2013.
- [15] M. Fernández-Martínez, M. A. Sánchez-Granero, and J. E. T. Segovia, "Measuring the self-similarity exponent in Lévy stable processes of financial time series," *Physica A: Statistical Mechanics and Its Applications*, vol. 392, no. 21, pp. 5330–5345, 2013.
- [16] S. Rostek and R. Schöbel, "A note on the use of fractional Brownian motion for financial modeling," *Economic Modelling*, vol. 30, pp. 30–35, 2013.
- [17] K. Domino, "The use of the Hurst exponent to investigate the global maximum of the Warsaw Stock Exchange WIG20 index," *Physica A: Statistical Mechanics and Its Applications*, vol. 391, no. 1–2, pp. 156–169, 2012.
- [18] I. Z. Rejichi and C. Aloui, "Hurst exponent behavior and assessment of the MENA stock markets efficiency," *Research in International Business and Finance*, vol. 26, no. 3, pp. 353–370, 2012.
- [19] J. Gao, J. Hu, X. Mao, and M. Perc, "Culturomics meets random fractal theory Insights into long-range correlations of social and natural phenomena over the past two centuries," *Journal of the Royal Society Interface*, vol. 9, no. 73, pp. 1956–1964, 2012.
- [20] A. M. Petersen, J. N. Tenenbaum, S. Havlin, H. E. Stanley, and M. Perc, "Languages cool as they expand: allometric scaling and the decreasing need for new words," *Scientific Reports*, vol. 2, no. 943, 2012.
- [21] M. Perc, "Evolution of the most common English words and phrases over the centuries," *Journal of the Royal Society Interface*, vol. 9, no. 77, pp. 3323–3328, 2012.
- [22] M. Perc, "Self-organization of progress across the century of physics," *Scientific Reports*, vol. 3, no. 1720, 2013.
- [23] E. N. Bruce, *Biomedical Signal Processing and Signal Modeling*, John Wiley & Sons, New York, NY, USA, 2001.
- [24] N. Sarkar and B. B. Chaudhuri, "An efficient approach to estimate fractal dimension of textural images," *Pattern Recognition*, vol. 25, no. 9, pp. 1035–1041, 1992.
- [25] S. S. Chen, J. M. Keller, and R. M. Crownover, "On the calculation of fractal features from images," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 15, no. 10, pp. 1087–1090, 1993.
- [26] N. Sarkar and B. B. Chaudhuri, "An efficient differential box-counting approach to compute fractal dimension of image," *IEEE Transactions on Systems, Man and Cybernetics*, vol. 24, no. 1, pp. 115–120, 1994.
- [27] X. C. Jin, S. H. Ong, and Jayasooriah, "A practical method for estimating fractal dimension," *Pattern Recognition Letters*, vol. 16, no. 5, pp. 457–464, 1995.
- [28] K. Falconer, *Fractal Geometry: Mathematical Foundations and Applications*, John Wiley & Sons, New York, NY, USA, 1990.
- [29] T. Lundahl, W. J. Ohley, S. M. Kay, and R. Siffert, "Fractional Brownian motion: a maximum likelihood estimator and its application to image texture," *IEEE Transactions on Medical Imaging*, vol. 5, no. 3, pp. 152–161, 1986.
- [30] P. Flandrin, "On the spectrum of fractional Brownian motions," *IEEE Transactions on Information Theory*, vol. 35, no. 1, pp. 197–199, 1989.
- [31] B. B. Mandelbrot and J. W. van Ness, "Fractional Brownian motions, fractional noises and applications," *SIAM Review*, vol. 10, no. 4, pp. 422–437, 1968.
- [32] Y.-C. Chang and S. Chang, "A fast estimation algorithm on the hurst parameter of discrete-time fractional Brownian motion," *IEEE Transactions on Signal Processing*, vol. 50, no. 3, pp. 554–559, 2002.
- [33] S.-C. Liu and S. Chang, "Dimension estimation of discrete-time fractional Brownian motion with applications to image texture classification," *IEEE Transactions on Image Processing*, vol. 6, no. 8, pp. 1176–1184, 1997.
- [34] M. S. Taqqu, V. Teverovsky, and W. Willinger, "Estimators for long-range dependence: an empirical study," *Fractals*, vol. 3, no. 4, pp. 785–798, 1995.
- [35] J. Beran, *Statistics for Long-Memory Processes*, Chapman & Hall, New York, NY, USA, 1994.
- [36] Y.-C. Chang, L.-H. Chen, L.-C. Lai, and C.-M. Chang, "An efficient variance estimator for the Hurst exponent of discrete-time fractional Gaussian noise," *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, vol. E95-A, no. 9, pp. 1506–1511, 2012.
- [37] Y.-C. Chang, L.-C. Lai, L.-H. Chen, C.-M. Chang, and C.-C. Chueh, "A Hurst exponent estimator based on autoregressive

- power spectrum estimation with order selection,” *Bio-Medical Materials and Engineering*, vol. 24, no. 1, pp. 1041–1051, 2014.
- [38] S. M. Kay, *Modern Spectral Estimation: Theory & Application*, Prentice-Hall, Englewood Cliffs, NJ, USA, 1988.
- [39] S. Haykin, *Modern Filters*, Macmillan, New York, NY, USA, 1989.
- [40] G. Samorodnitsky and M. S. Taqqu, *Stable Non-Gaussian Random Processes*, Chapman & Hall, New York, NY, USA, 1994.
- [41] R. J. Schilling and S. L. Harris, *Applied Numerical Methods for Engineers: Using MATLAB and C*, Brooks/Cole, New York, NY, USA, 2000.
- [42] Y.-C. Chang, “N-dimension golden section search: its variants and limitations,” in *Proceedings of the 2nd International Conference on Biomedical Engineering and Informatics (BMEI '09)*, pp. 1–6, Tianjin, China, October 2009.
- [43] S. M. Kay, *Fundamentals of Statistical Signal Processing: Estimation Theory*, Prentice-Hall, Englewood Cliffs, NJ, USA, 1993.



Hindawi

Submit your manuscripts at
<http://www.hindawi.com>

