Multiscale Higuchi's fractal dimension method

A. Yilmaz [ORCID] · **G. Unal**

Abstract The aim of this study is to introduce a new method for the evaluation of complexity properties of time series by extending Higuchi's fractal dimension (HFD) over multiple scales. Multiscale Higuchi's fractal dimension (MSHG) is presented and demonstrated on a number of stochastic time series and chaotic time series, starting with the examination of the selection of the effective scaling filter among several widely used filtering methods and then diving into the application of HFD through the scales obtained by coarse-graining procedure. Moreover, on the basis of MSHG, fractal dimension and Hurst exponent relationship are studied by employing MSHG method with computation of Hurst value in multiple scales, simultaneously. Consequently, it is found that the proposed method, MSHG produces remarkable results by exposing unique complexity features of time series in multiple scales. It is also discovered that MSHG with multiscale Hurst exponent calculation leads to revelation of distinguishing patterns between verifying stochastic time series and diverging chaotic time series. In light of these findings, it can be inferred that the proposed methods can be utilized for the characterization and classification of time series in terms of complexity.

A. Yilmaz (✉)
Yeditepe University, 34755 Istanbul, Turkey
e-mail: adil.yilmaz@std.yeditepe.edu.tr

G. Unal
Bahcesehir University, 34349 Istanbul, Turkey
e-mail: gazanfer.unal@eas.bau.edu.tr

1 Introduction

Fractal phenomenon, since it was first introduced by Mandelbrot [1], has been attracting widespread interest. Fractal characteristic basically is based on self-similarity and refers to complexity of a system which is represented by the fractal dimension. The fractal dimension has found application in analysis of time series and determination of the nonlinear dynamical properties. As an example, it is used in bioscience for the determination of anomalies of the human body [2–4], in physics for the examination of solar activity [5], in atmospheric research for the analysis of rainfall data series [6], in mechanical engineering for damage detection of steel beam [7], in materials science for the measurement of silicon content in pig iron [8] and for assessing structural properties of materials [9] and in finance for the analysis of stock market indices [10].

There are number of algorithms used to calculate fractal dimension. Among them, especially, Higuchi's algorithm [11] comes forward for its simplicity and efficiency. In 1988, Higuchi introduced his algorithm which approximates the fractal dimension directly from time series by means of the length of the irregular curve. So, in comparison with other methods especially that reconstructs the attractor phase space, Higuchi's algo-

rithm runs faster and can be applied to shorter time series for the estimation of the dimensional complexity of time series.

In this work, we extend Higuchi's fractal dimension (HFD) analysis from monoscale to multiscale. In 2005, Costa et al. [12] introduced the multiscale complexity measure for biological signals. According to Costa, because of complexity of biological systems, they need to work across multiple spatial and temporal scales. So, they developed multiscale entropy (MSE) analysis by combining sample entropy as a complexity measure and coarse-graining procedure as a scaling filter.

Since then, multiscale analysis have been enriching by the works of scientists who noticed the benefits of observing information in multiple scales. Entropy-based multiscale evaluations have been the main focus in these works. A number of versions of multiscale permutation entropy have been applied for the observation of the dynamical characteristics of EEG data [13–15]. Modified version for multiscroll chaotic systems was introduced [16], and modified multiscale entropy for short term time series was developed [17]. More recently, improved version of multiscale permutation entropy [18] and multiscale transfer entropy were proposed [19].

We apply the same idea for HFD in order to investigate how complexity features of time series change through multiple scales. For the choice of scaling filter, several useful filter options were tested and the mean filter as the coarse-graining procedure come forward among them as the most efficient one. As a result, we introduce multiscale Higuchi's fractal dimension (MSHG) analysis by putting together coarse-graining procedure as a scaling filter and HFD as a complexity measure. Then, we demonstrate the MSHG analysis on stochastic time series and chaotic time series.

We also examine how the relationship between complexity measure fractal dimension (D) and long-range dependence measure Hurst exponent (H) which is formulized as $D = 2 - H$ changes in multiscale. We again experiment MSHG on the same set of chaotic time series and stochastic time series. The results show that as the relationship in monoscale holds through multiple scales with MSHG for the stochastic time series, it diverges for the chaotic time series. Therefore, such distinguishing observations in multiscale based on H and D can be useful for characterizing time series whether they possess stochastic or chaotic properties.

The rest of the paper is organized as follows. In subsequent section, which filtering method, how and why chosen is given in detail. All examined methods are briefly explained, and the comparative results are presented. Then, in Sect. 2, the most efficient method, mean filter and Higuchi's fractal dimension algorithm which comprise MSHG are elaborated. In Sect. 3, MSHG is demonstrated on selected chaotic time series and stochastic time series extensively. Section 4 looks into the relationship between D and H in multiple scales in a fresh way by conducting MSHG and Hurst algorithms in parallel in successively scaled group of time series. In the final section, all findings are summarized with concluding comments.

2 Methodology

The proposed method for the analysis of complex characters of time series that we call it multiscale Higuchi's fractal dimension (MSHG) incorporates a scaling part and the complexity measurement part. For the scaling part, the coarse-graining algorithm is the most used in the multiscale literature although there are many filters especially employed in image processing. In this section, we firstly look at some good candidates among these filters by giving brief instructions and comparing their effectiveness on all chaotic and stochastic time series under examination. Gaussian filter, Wiener filter, mean shift filter, bilateral filter, total variation filter, standard deviation filter, max filter, harmonic mean filter and gradient filter are the filters examined as an alternative to mean filter.

2.1 Scaling filters

2.1.1 Coarse-graining procedure

Coarse-graining procedure or the mean filter is a moving average process. It is applied to time series in an order by using a scale factor τ with a low-pass filter. In our study, overlapping window is used to minimize data loss to a single one in each step. As a result of windowing, in each step, data series change. The information in each interval which is related to other intervals can be captured by each window.
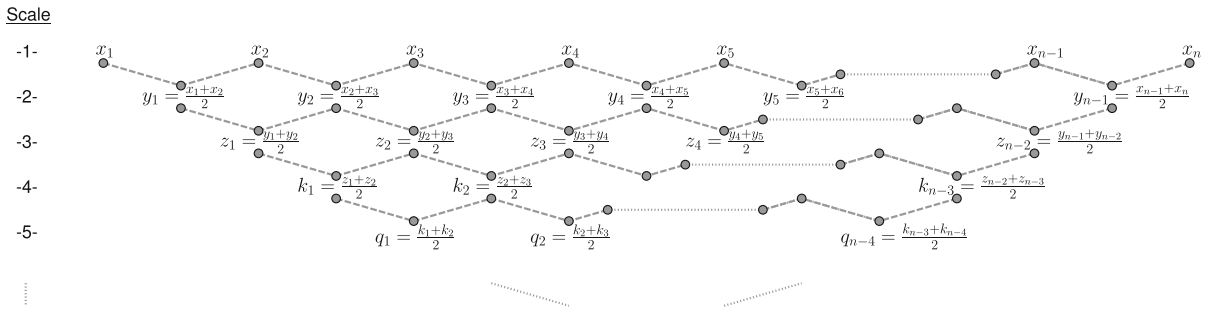
Scale

Fig. 1 Multiscale overlapping coarse-graining algorithm

For one-dimensional time series of $x_1, x_2, \ldots, x_N$, coarse-graining procedure is described as

$$y_{n,\tau} = \frac{1}{\tau} \sum_{i=0}^{\tau-1} x_{n+i}, \tag{1}$$

where, consecutively, n, N, i and τ denote the subscript of coarse-grained data series, the length of data series, the loop key and the scale factor. The length of each coarse-grained time series becomes $N - \tau$. Equation 1 gives each data point on each scale. The procedure is also given with an illustration describing it visually in Fig. 1. As scale 1 is the original time series, at each scale, values of consecutive data pairs are averaged to obtain the value of each point of subsequent scale. As a consequence of this downsampling, the length of data is shortened on each scale. However, it keeps the loss of data at minimum by reducing the length of the series only by one which serves to transfer more information through scales, comparison with other coarse-graining procedures which consume more number of data points.

2.1.2 Gaussian filter

Gaussian filter has got many applications, especially in image processing. In Gaussian filter, simply, the average of weighting values replace the intensity value of the pixel and its neighbor pixels. The Gaussian filter use convolution of required Gaussian function g. This function that is governed by the variance σ^2 is described as

$$g(x, y, \sigma) = \frac{1}{2\pi\sigma^2} e^{-(\frac{x^2+y^2}{2\sigma^2})}, \tag{2}$$

where x and y are the distances in the horizontal axis and vertical axis consecutively. Equation 2 is used to estimate the coefficients for a Gaussian template and

then convolved [20]. The Gaussian filter can be applied in one or more dimensions. The advantage of the Gaussian filter compared to direct averaging is the enhanced performance as a result of maintaining more features.

2.1.3 Wiener filter

The Wiener filter is actually a linear estimation of a signal. It is, especially advantageous while working with noisy signals. Therefore, it finds wide-range applications for linear prediction, signal restoration and system identification. For an original signal x and an additive noise n, the Wiener filter is described as

$$W(u, v) = \frac{H^*(u, v) S_{xx}(u, v)}{\mid H(u, v) \mid^2 S_{xx}(u, v) + S_{nn}(u, v)}, \tag{3}$$

where u, v are the location parameters of frequency. $H(u, v)$ is blurring or degradation filter, and H^* is its conjugate. $S_{xx}(u, v)$ denotes the power spectrum of the original signal which is computed by the Fourier transform of the signal autocorrelation. $S_{nn}(u, v)$ is the power spectrum of the additive noise which is acquired by the Fourier transform of the noise autocorrelation [21].

2.1.4 Mean shift filter

Mean shift filtering is primarily related to data clustering. It has many applications in the areas of computer vision, smoothing, segmentation and tracking. As to Fukunaga's introduction, the mean shift vector is

$$m(x) = \frac{\sum_{i=1}^{n} g(\frac{x-x_i}{h}) x_i}{\sum_{i=1}^{n} g(\frac{x-x_i}{h})} - x, \tag{4}$$

where h is bandwidth radius, n is data point in observations x_i where $i = 1, \ldots, n$ in d-dimension space R^d and $g(x) = -K(x)$. Here, K is a kernel (for

instance Gaussian kernel which is the most popular one) employed in order to estimate probability density. The algorithm works by iteratively calculating the mean of a window around a data point as shifting the center of the window until the convergence. The mean shift vector is computed until the convergence for each point x_i according to a selected search window. The algorithm looks for a local maximum of density of a distribution [22,23].

2.1.5 Bilateral filter

Bilateral filter operates as a smoothing filter by replacing each point with the nonlinear combination of the neigbouring values. Its applications can be found in denoising, optical-flow estimation, texture editing and so on. As I denotes an image and p, q represent some pixel positions, bilateral filter is described as

$$BF[I]_p = \frac{1}{W_p} \sum_{q \in S} G_{\sigma_s}(\|p - q\|) G_{\sigma_r}(I_p - I_q) I_q.$$
(5)

The normalization factor W_p in Eq. 5 is

$$W_p = \sum_{q \in S} G_{\sigma_s}(\|p - q\|) G_{\sigma_r}(I_p - I_q),$$
(6)

and the two-dimensional Gaussian kernel $G_\sigma(x)$ in Eq. 6 is

$$G_\sigma(x) = \frac{1}{2\pi\sigma^2} \exp\left(\frac{-x^2}{2\sigma^2}\right).$$
(7)

So, based on Eq. 7, G_{σ_s} is the Gaussian kernel associated with location which decreases the effects of far points and G_{σ_r} is the Gaussian related to value which decreases the effects of points q with an intensity value different from I_p [24,25].

2.1.6 Total variation filter

The total variation algorithm was first introduced for image denoising and reconstruction [26]. For a signal x with an additive noise n which is observed in the form of $y = x + n$, in order to estimate x, total variation filtering measures the amount of changes between signal values and is described as the minimization of following formula:

$$J(x) = \|y - x\|_2^2 + \lambda\|Dx\|_1,$$
(8)

where λ denotes the regularization parameter. The L1 norm matrix $\|Dx\|_1$ given in Eq. 8 can also be expressed as

$$\sum_{n=2}^{N} |x(i) - x(i - 1)|,$$
(9)

for $1 \leq i \leq N$ for N-point signal $x(i)$ in Eq. 9 [27].

2.1.7 Standard deviation filter

As the name suggests, with the standard deviation filter, the standard deviation of the data points in a particular range neighborhood is used. Then, these are returned to the place of every data point. The standard deviation formula is given below as

$$SD = \sqrt{\frac{\sum(x_i - \bar{x})^2}{r * c - 1}},$$
(10)

where x_i denotes the value of particular pixel and $\bar{x}$ denotes the mean of the pixel values in the filter range. Besides, r and c are the size of the filter in rows and columns, respectively [28].

2.1.8 Max and Min filter

Max filter and min filter are classified among nonlinear filters. They filter an image by refining only the minimum or maximum of all pixels in a local region $R_{u,v}$ of an image. Each pixel in an image is assigned a new value equal to the maximum or minimum value in a neighborhood around itself. This process is summarized as

$$I'(u, v) \leftarrow \min\{I(u + i, v + j) \mid (i, j) \in R\}, \quad (11)$$
$$I'(u, v) \leftarrow \max\{I(u + i, v + j) \mid (i, j) \in R\}, \quad (12)$$

where R denotes the filter region, I and I' denote the original image and the filtered image, u and v denote the position parameters. Algorithms replace every value in time series by the maximum or minimum in a determined range [29].

2.1.9 Harmonic mean filter

Harmonic mean filter, in essence, is a different version of mean filter. It is quite useful for the removal of Gaussian noise as well as preserving edge features. For two dimensional space, the harmonic mean filter is given as

$$HM(I) = \frac{mn}{\sum_{(i,j) \in W} \frac{1}{I(x+i, y+j)}},$$
(13)

where x and y denote coordinates over the image and I, i and j denote the coordinates in a window W with the size of m, n which are the length of each dimension [30,31]. The algorithm works as replacing every value by the harmonic mean value in a determined range.

2.1.10 Gradient filter

The gradient of the function I at position (u, v) is given as a vector:

$$\nabla I(u, v) = \begin{pmatrix} I_x(u, v) \\ I_y(u, v) \end{pmatrix} = \begin{pmatrix} \frac{\partial I}{\partial x}(u, v) \\ \frac{\partial I}{\partial y}(u, v) \end{pmatrix}. \qquad (14)$$

Basically, the partial derivatives of horizontal and vertical lines constitute the gradient function merely consists of. There are the horizontal and vertical gradient filters respond to swift changes in horizontal axis and the vertical axis, respectively [29].

Gradient filter based on the vector given in Eq. 14 calculates the magnitude of the gradient of an image which is the rate of increase and described as

$$|\nabla I| = \sqrt{\left(\frac{\partial I}{\partial x}(u, v)\right)^2 + \left(\frac{\partial I}{\partial y}(u, v)\right)^2}. \qquad (15)$$

Because the magnitude does not vary when the image is rotated or oriented in a different position, it is especially used in edge detection.

2.1.11 Findings on scaling filters

All scaling filters selected for this study are the member of low pass filter family. Among these filters, while mean filter defined in Eq. 1 is one of the most used filter in practice, Gaussian filter which exercises the convolution of Gaussian function given in Eq. 2 is also very popular in image processing applications. The Wiener filter described in Eq. 3 as a dependent of the power spectrum of the signal and additive noise with a degradation filter is preferred in linear prediction. Mean shift filter whose algorithm is summarized in Eq. 4 computes the mean of a window until the convergence. Bilateral filter smoothes images while preserving edges according to Eq. 5. Total variation filter is a slope-preserving method which searches for the minimum of Eq. 8. Standard deviation filter based on Eq. 10 is employed by computing standard deviation of data points in the filter range. Max and min filter presented in Eqs. 11 and 12 basically calculate the maximum and minimum values of data points in filter region. Harmonic mean filter

as modification of mean filter operates by replacing data points in a local region by the value calculated by Eq. 13. In gradient filter algorithm, the magnitude of the gradient given in Eq. 15 based on the partial derivatives of horizontal and vertical lines is computed.

For the examination of these filters, built-in functions of Mathematica v11.0.1.0 are used. Results are presented in a way to allow comparisons. Because almost each pair result is closely identical, only figures for fractional Brownian motion (fBm) are given in Fig. 2 to avoid repetition and for clear presentation. As the data sets, five different fBm processes are generated with Hurst exponent values of 0.25, 0.40, 0.50, 0.60 and 0.75 and represented with different colors for providing distinguishable observations.

Figure 2 consists of ten subfigures. Each subfigure belongs to each filter in the list of scaling filters under investigation. These subfigures show fractal dimension value on y axis and scale value on x axis. The number of scales on which D values are calculated and plotted is 12.

As observed in subfigures of Fig. 2, mean filter, Gaussian filter, mean shift filter, bilateral filter, total variation filter, standard deviation filter produce similar patterns albeit with different values in each scale. It is not really clear to determine the most efficient scaling filter because of such close patterns. However, the most used filter, mean filter and Gaussian filter are applied to upcoming analysis side by side and observations show us mean filter is slightly more consistent for all the data series. So, the choice of scaling filter for the rest of our calculation arises as mean filter.

Although the mean filter acts slightly better with Higuchi's algorithm, it is not easy to generalize it to other multiscale methods. In our previous studies, for example, in the application of correlation dimension algorithm in multiscale, Gaussian filter had generated more stable results than mean filter as well as other filters tested. Therefore, for now, for each algorithm, when analysis is made in multiscale, the choice of scaling filter is better to be made after testing variety of filters until developing an efficient uniform method for this purpose. For not diverging from the main purpose of this study, this subject of the choice of scaling filter is not presented in detail any further.
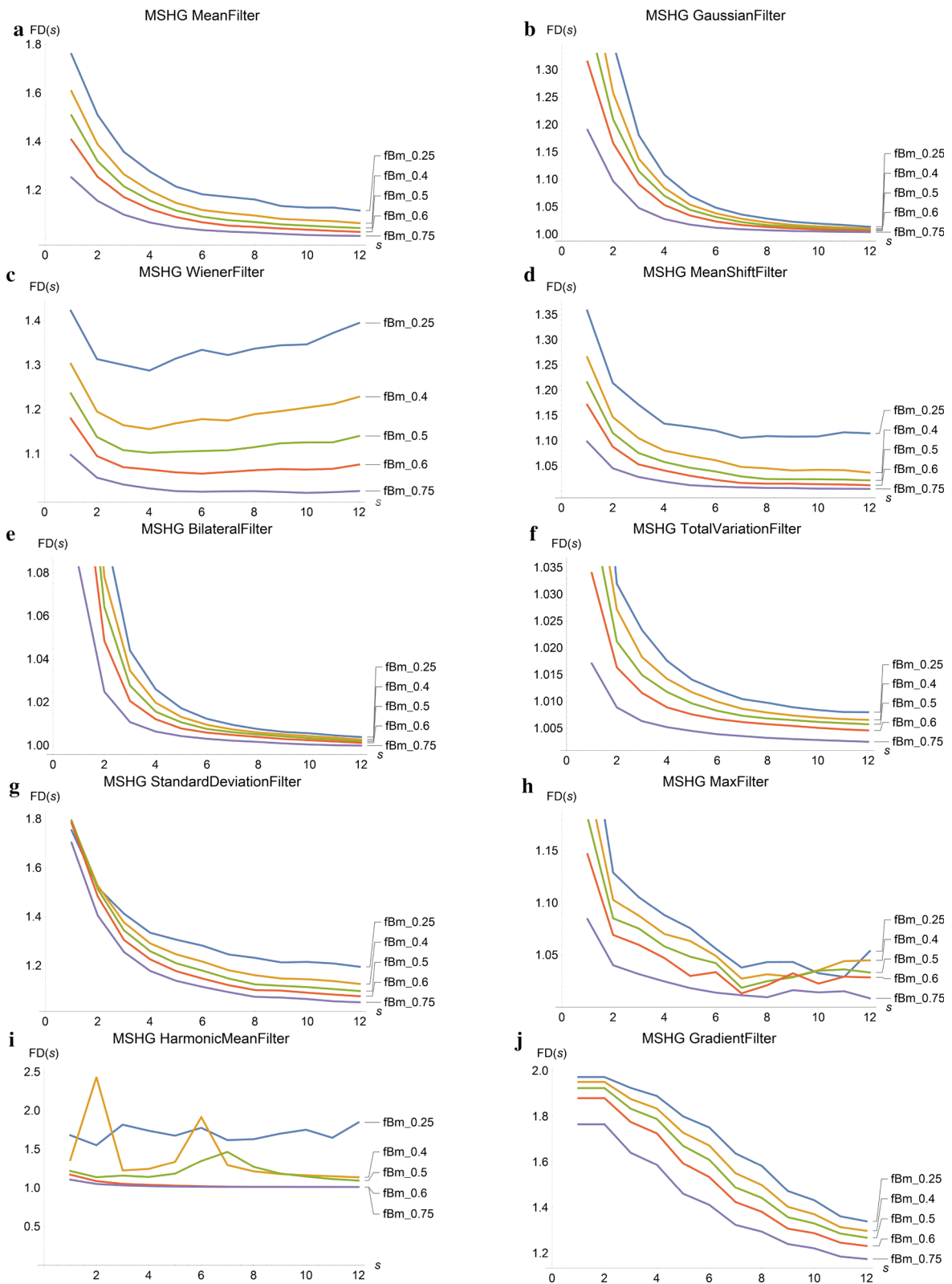
Fig. 2 MSHG results of all filters. **a** Mean filter, **b** Gaussian filter, **c** Wiener filter, **d** Mean shift filter, **e** Bilateral filter, **f** Total variation filter, **g** Standard deviation filter, **h** Max filter, **i** Harmonic mean filter, **j** Gradient filter

2.2 Fractal dimension

Complex interlinked systems like stock markets, human heart, neural structures, the digital networking systems are generally made up of multiple subsystems governed hierarchically show nonlinear deterministic characteristics and stochastic characteristics. A complex system can be examined for learning about its behavior by measuring its particular signals that indicate nonlinearity, sensitivity to initial conditions, long memory, severe volatility and nonstationarity [32].

Fractal theory gives effective method for characterizing complex structure of such systems. Fractals are interpreted by a non-integral dimension named as fractal dimension. The fractal phenomenon is found everywhere and studied in many fields of science like in finance for the analysis of price variations [33] and stock markets [34], in physics for the detection of periodic components in seismograms [35] or in engineering for porous media [36] and so on.

Fractal structure is characterized with self-similarity, and its complexity is measured by its fractal dimension which is easier to be computed from data sets. There are numerous methods for the measurement of fractal dimension like Higuchi, Kantz, Maragos and Sun or Burlaga and Klein and so on. Among them, Higuchi's fractal dimension (HFD) is a fast nonlinear computational method which yields more accurate result in comparison with others [37]. Shifts in the structure of time series in a time domain over a specific characteristic frequency make it hard to find out power law indices and a characteristic time scale from the power spectrum. Stable indices and time scale related to the characteristic frequency can be provided by the HFD method even there are very limited data points available [11].

2.3 Higuchi's fractal dimension algorithm

Higuchi's fractal dimension algorithm is described as follows [11]. Giving a N-length one-dimensional time series with equal intervals $x(1), x(2), \ldots, x(n)$, a new time series X_k^m is constructed as

$$X_k^m : x(m), x(m + k), x(m + 2k),$$
$$\ldots, x\left(m + int\left[\frac{N - m}{k}\right].k\right), \quad (16)$$

where k as an integer is the time interval and number of new time series sets and $m = 1, 2, \ldots, k$. Therefore,

Eq. 16 gives k number of new time series. The length of each new time series obtained by X_k^m is defined as

$$L_m(k) = \left(\frac{1}{k}\left\{\left[\sum_{i=1}^{\text{int}[\frac{N-m}{k}]} |x(m + ik) \right.\right.\right.$$
$$\left.\left.\left. -x(m + (i - 1).k)|\right] \frac{N - 1}{\text{int}[\frac{N-m}{k}.k]}\right\}\right), \quad (17)$$

where $\frac{N-1}{\text{int}[\frac{N-m}{k}]}$ is the normalization factor for the curve length of k sets of constructed time series. By Eq. 17, the length of the curve for the interval k is obtained by computing the average value over k series of $L_m(k)$ as

$$L(k) = \frac{1}{k}\sum_{m=1}^{k} L_m(k). \quad (18)$$

Then, based on Eq. 18, the fractal dimension D_f is described by

$$L(k) \sim k^{D_f}. \quad (19)$$

So, the complexity measure D_f can be calculated by the least-squares linear best fit procedure as finding the slope of the curve on the graph of $\ln(L(k)) - \ln(1/k)$ based on Eq. 19. D_f takes values ranging between 1 and 2.

In our study, HFD, a measure of self-similarity and complexity of time series, is extended to multiple scales which is provided by coarse-graining scaling filter. So, new procedure named MSHG allows uncovering different characteristics which helps to understand and identify the nature of time series under examination. To do this, HFD value of each scale gathered by scaling filter is calculated. Then, all HFD values in y axis versus scale number in x axis are plotted. The pattern of the plot and values shows the particular characteristics of different time series. In the following section, on various time series data sets in different classes, namely stochastic and chaotic time series, MSHG is demonstrated in detail.

3 Applications

3.1 Stochastic time series

Applications and demonstrations on specific time series of MSHG algorithm start in this section with stochastic time series of white noise, fractional Brownian motion
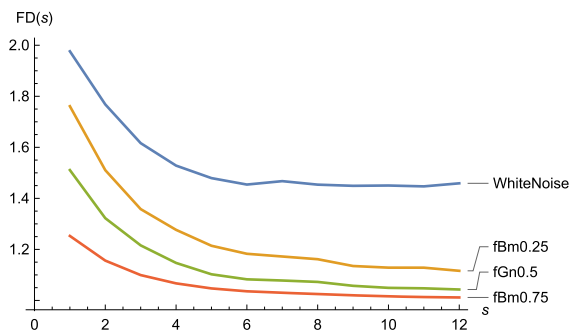
Fig. 3 MSHG results of stochastic processes

(fBm) and fractional Gaussian noise (fGn) generated by Mathematica's related process functions.

As fBm and fGn are self-similar stochastic processes with long-range dependence, white noise is a random noise. fGn and fBm are related processes since fGn is the increments of fBm. fBm is a Gaussian process with mean function μt and its covariance function is written as

$$\gamma(t, s) = \frac{\sigma^2 (s^{2H} + t^{2H} - |t - s|^{2H})}{2}. \tag{20}$$

Fractional Gaussian noise process is also a Gaussian noise with mean function μ and covariance function

$$\gamma(t, s) = \frac{\sigma^2 (|t - s - 1|^{2H} - 2|t - s|^{2H} + |t - s + 1|^{2H})}{2}, \tag{21}$$

where σ is volatility and H is the Hurst exponent $H \in (0, 1)$.

The Hurst exponent quantifies the Hurst phenomenon which describes the long-range dependence of the fBM given in Eq. 20 and fGn defined in Eq. 21 [38].

If H is set to 0.5, the process is Brownian motion and independently distributed. When H is different from 0.5, the observations are not independent and system is short-term memory process if $H < 0.5$ and long-term memory process if $H > 0.5$.

As a set of stochastic time series, white noise, fGn ($H = 0.5$) and fBm ($H = 0.25, 0.75$) processes are exercised with the length of 1250 time steps. The results are presented in Fig. 3 which shows similar patterns for every time series in different ranges.

3.2 Financial time series

MSHG is continued to be tested on stochastic time series particularly with financial time series processes

in this section. For this purpose, two important processes of FARIMA and FIGARCH are utilized.

3.2.1 FARIMA

Autoregressive fractionally integrated moving average model (FARIMA), in a similar sense to FIGARCH, is a modification of autoregressive process (AR) and moving average process (MA) models allowing fractional differencing.

MA(q) model is written as

$$X_t = \mu + \sum_{i=1}^{q} \theta_i \epsilon_{t-i} + \epsilon_t. \tag{22}$$

And AR(p) is given as

$$X_t = \mu + \sum_{i=1}^{p} \phi_i X_{t-i} + \epsilon_t, \tag{23}$$

where $\mu = E[y_t]$, θ and ϕ are the parameters of the MA model and AR model consecutively. ϵ is the white noise process with the properties of $E(\epsilon_t) = 0$ and $var(\epsilon_t) = \sigma^2$ [39].

ARIMA(p, d, q) as a combination of AR(p) defined in Eq. 23 and MA(q) given in Eq. 22 models is described as follows [40]:

$$\left(1 - \sum_{i=1}^{p} \phi_i B^i\right) (1 - B)^d (X_t - \mu)$$
$$= \left(1 + \sum_{i=1}^{q} \theta_i B^i\right) \epsilon_t, \tag{24}$$

where $(1 - B)^d$ is the difference operator as d takes integer values. B represents the backshift operator. It works in the notation as $B^i X_t = X_{t-i}$.

So, ARIMA models given in Eq. 24 are especially powerful when it comes to short-range dependence [41]. And, when d is allowed to take fractional values, it is suggested that the model becomes better at capturing long-range dependence [42]. Then, as denoting the autoregressive order, the difference coefficient and the moving average order with p, d and q, in general form FARIMA(p, d, q) process can be described as

$$\phi(B)(1 - B)^d X_t = \theta(B)\epsilon_t, \tag{25}$$

where $d \in (-0.5, 0.5)$, $\phi(B) = 1 - \phi_1 B - \cdots - \phi_p B^p$, $\theta(B) = 1 + \theta_1 B + \cdots + \theta_q B^q$ and $(1 - B)^d$ is the fractional difference operator.

3.2.2 FIGARCH

Fractionally integrated generalized autoregressive conditional heteroskedastic (FIGARCH) process is a class of GARCH process with more persistence on the conditional variance which allows estimation long memory of conditional volatility.

The GARCH model allows the conditional variance to be dependent upon previous own lags. The GARCH(p, q) is

$$\sigma_t^2 = \alpha_0 + \sum_{i=1}^{q} \alpha_i \epsilon_{t-i}^2 + \sum_{j=1}^{p} \beta_j \sigma_{t-j}^2, \tag{26}$$

where $\alpha_i \geq 0$ and $\beta_j \geq 0$ are the parameters and σ_t^2 is the conditional variance. The conditional variance of error term of a model ϵ_t with the properties of $\epsilon_t \sim N(0, \sigma_t^2)$ is written as

$$\sigma_t^2 = var(\epsilon_t|\epsilon_{t-1}, \epsilon_{t-2}, \ldots)$$
$$= E[(\epsilon_t)^2|\epsilon_{t-1}, \epsilon_{t-2}, \ldots]. \tag{27}$$

The GARCH(p, q) model given in Eq. 26 can be expressed in a form that shows that it is effectively an ARMA(m, p) model for the conditional variance as

$$\epsilon_t^2[1 - \alpha_1(B) - \beta_1(B)] = \alpha_0 + [1 - \beta_1(B)]v_t, \tag{28}$$

where v_t is mean zero serially uncorrelated

$$v_t = \epsilon_t^2 - \sigma_t^2, \tag{29}$$

and $m \equiv \max\{p, q\}$. The GARCH process is defined to be integrated in variance. So, based on Eqs. 27, 28 and 29, IGARCH can be written in the same notation as

$$[1 - \alpha_1(B) - \beta_1(B)](1 - B)\epsilon_t^2$$
$$= \alpha_0 + [1 - \beta_1(B)]v_t, \tag{30}$$

when,

$$\sum_{i=1}^{p} \beta_i + \sum_{j=1}^{q} \alpha_j = 1. \tag{31}$$

If fractional difference operator d is added to IGARCH(p, q) given in Eq. 30 with the condition of Eq. 31, then FIGARCH(p, d, q) is obtained and described as

$$\phi(B)(1 - B)^d u_t^2 = \omega + [1 - \beta(B)]v_t, \tag{32}$$

where $0 < d < 1$ and $\phi(B) \equiv [1 - \alpha_1(B) - \beta_1(B)]$ is of order $m - 1$ [43].

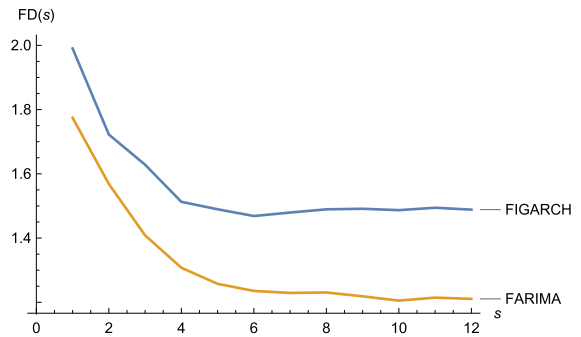FIGARCH process has got fractional and long memory properties but does not possess chaotic properties

Fig. 4 MSHG results of FIGARCH and FARIMA

[44]. Identifying these properties is useful considering that MSHG results of the chaotic times series are compared with stochastic time series in the subsequent section.

3.2.3 MSHG results of FIGARCH and FARIMA

For the application of MSHG on FIGARCH(p, d, q) in Eq. 32 and FARIMA(p, d, q) in Eq. 25, sample data series for both processes are generated with the length of 1250 data points, and then, figures of HFD vs. number of scales are formed as presented in Fig. 4. FIGARCH and FARIMA lines follow the same pattern as it was in MSHG results of white noise, fGn and fBm presented in previous section. These similar pattern although occurring in various values may suggest that it is a property of stochastic time series in multiscale. In the next section, MSHG is run on several chaotic time series and its results are compared with the findings in this section whether or not these time series with different characteristics produce unique patterns.

3.3 Chaotic systems

A chaotic system is a complex dynamic nonlinear deterministic system which is unpredictable in the long term because of its sensitivity to changes of initial conditions. Even the smallest change at one point can cause a very large shift in future point as a result of information transmission through following data points. There are various methods and algorithms like HFD to measure the complexity of such characteristic systems in monoscale. In this section, MSHG is applied to examine the feature of complexity of chaotic time series in

multiscale. The chaotic time series data set we use comprises three chaotic time series which are generated from two-dimensional chaotic discrete maps of Henon map, Duffing map and Ikeda map.

3.3.1 Henon map

Michel Henon introduced Henon map in 1976 with

$$x_{n+1} = 1 - ax_n^2 + y_n, \tag{33}$$
$$y_{n+1} = bx_n, \tag{34}$$

where a, b parameters are given the values of 1.4 and 0.3 to obtain chaotic Henon time series for the calculation [45].

3.3.2 Duffing map

By assigning 2.75 and 0.2 values to a and b parameters in

$$x_{n+1} = y_n, \tag{35}$$
$$y_{n+1} = -bx_n + ay_n - y_n^3, \tag{36}$$

Duffing chaotic time series are acquired [46].

3.3.3 Ikeda map

These three equations,

$$x_{n+1} = 1 + u(x_n \cos(z) - y_n \sin(z)), \tag{37}$$
$$y_{n+1} = u(x_n \sin(z) + y_n \cos(z)), \tag{38}$$
$$z = 0.4 - \left(\frac{6}{1 + x_n^2 + y_n^2} \right), \tag{39}$$

govern the Ikeda map [47]. Time series can be generated by setting u parameter to 0.918.

The result of MSHG calculations of these three chaotic time series generated by using Eqs. 33, 34, 35, 36, 37, 38 and 39 with the length of 1250 data points are summarized in Fig. 5. Lines emerging on D-Scale plane seem quite irregular and leading through uncertain directions. These patterns are very different from the ones revealed with stochastic time series. In comparison with stochastic time series, more irregular and jagged lines through consecutive scales of the
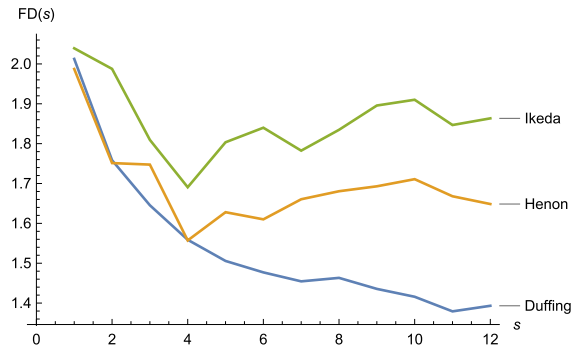

Fig. 5 MSHG results of chaotic maps

chaotic time series figure take the place of smooth and horizontal lines of the stochastic time series.

4 HFD and Hurst exponent link in multiscale

The aim of this section is to investigate the relationship between HFD and H in multiscale. Through previous sections, MSHG method has been introduced and demonstrated on time series with different characteristics. With MSHG method, HFD is expanded on multiple scales and its distinguishing revelations are presented. In a similar sense, Hurst exponent calculations are exercised on multiple scales numerically, in order to examine the link between H and HFD also holds in multiple scales.

4.1 Hurst exponent

Hurst exponent (H) as a measure of long memory dependence has also been studied and applied in wide range of different fields such as in hydrology [48], in medicine and biology [49], in astrophysics [50], in finance [51] and so on. Hurst exponent was introduced in 1951 by H. E. Hurst while investigating river flows of the Nile basin [52]. The method developed was called rescaled range or R/S analysis. Letting X_n be the inflow of to the dam in the original problem or any time series in period $n = 1, 2, \ldots, k$, the rescaled adjusted range statistic is

$$R/S(k) = \frac{\max_{0 \leq j \leq k} \left\{ \sum_{n=1}^{j} X_n - \frac{j}{k} \sum_{n=1}^{k} X_n \right\} - \min_{0 \leq j \leq k} \left\{ \sum_{n=1}^{j} X_n - \frac{j}{k} \sum_{n=1}^{k} X_n \right\}}{\sqrt{\frac{1}{k} \sum_{j=1}^{k} \left(X_j - \frac{1}{k} \sum_{j=1}^{k} X_j \right)^2}}. \tag{40}$$

The numerator is called the adjusted range, and the denominator is the sample standard deviation in Eq. 40 where $j = 1, 2, \ldots, k$. After examination of many different time series, Hurst found that $R/S(k) \approx k^H c$ for large k where c denotes some constant [53].

Hurst exponent (H) takes values between 0 and 1. If $H = 0.5$, the system is independently distributed and Brownian motion. However, if H value is different from 0.5, then system possesses memory of previous points and no longer identified as independent. It is described as a short-term memory or anti-persistent system if $H < 0.5$ and long-term memory or persistent system if $H > 0.5$.

Mandelbrot later introduced a method which has been widely used. H is computed by Mandelbrot's method [54] described as

$$R/S_t = ct^H, \tag{41}$$

after taking logarithms of both sides, formula becomes linearized as,

$$\log(R/S_t) = \log(c) + H\log(t)), \tag{42}$$

where R/S_t is,

$$R/S_t = \left[1/M \sum_{m=1}^{M} \left(R_{I_m}/\sigma_{I_m} \right) \right] = ct^H, \tag{43}$$

$$R_{I_m} = \max \left(X_{k,m} \right) - \min \left(X_{k,m} \right), \tag{44}$$

$$X_{k,m} = \sum_{k=1}^{n} \left(N_{k,m} - \mu_{I_m} \right), \tag{45}$$

where, σ, μ, I_m, N, $N_{k,m}$, successively denote the standard deviation, the mean of M sub-periods for $m = 1, 2, \ldots, M$, each of the M subperiods, the number of points in time series, each element of a given time series. Also, as c represents some constant, $t = N/M$ and $k = 1, 2, \ldots, t$ [55].

Time series of N observations are divided to t length of M subperiods. Standard deviation and mean of each subperiods are calculated. Then, by calculation of variation R_{I_m} given in Eq. 44 dependent on $X_{k,m}$ calculated as in Eq. 45 provides to estimation of the mean R/S_t in Eq. 43 by R_{I_m}/σ_{I_m} of all subperiods. Finally, H value can be computed by a linear regression of Eq. 42 as a solution of relationship given in Eq. 41.

4.2 The relationship between fractal dimension and Hurst exponent

Mandelbrot [1] first introduced the relationship between Hurst exponent and fractal dimension as

$$D = 2 - H \tag{46}$$

where, H denotes the Hurst exponent and D denotes fractal dimension which is calculated by Higuchi's algorithm in this study and denoted with HFD.

The relationship described in Eq. 46 that theoretically relates HFD to H value is investigated in multiscale by making use of MSHG and calculating H value scale by scale simultaneously on the same stochastic time series and chaotic time series used in previous section.

This algorithm is used to find H values on each scale which is obtained by coarse-graining procedure. Number of scales are limited to 12. Each H value against its related scale is plotted as it is repeated for MSHG simultaneously. Besides, for each scale, sum of H and D values are calculated and plotted to observe how $D + H = 2$ relationship appears.

These calculations are made for all stochastic time series (white noise, fGn, fBm, FIGARCH, FARIMA) and chaotic time series (Henon, Duffing, Ikeda) sets used in previous section. Each stochastic time series is generated by computing mean path of numerous sample random processes. Plots obtained for each time series are presented on a single figure (Fig. 6) in order to provide efficient viewing. Subfigures are lined up starting from stochastic time series through financial time series to chaotic time series. In subfigures, when y axis shows D and H values, x axis represents the number of scale.

In Fig. 6, subfigures from a to e present the changes of H and D values individually as well as $D + H$ in multiscale for the stochastic time series. As it is clearly seen that even though H and D patterns are different for almost all stochastic time series in various degrees, $D + H$ value converge at the value of 2, especially after the scale 4 by supporting the relationship generalized by Eq. 46.
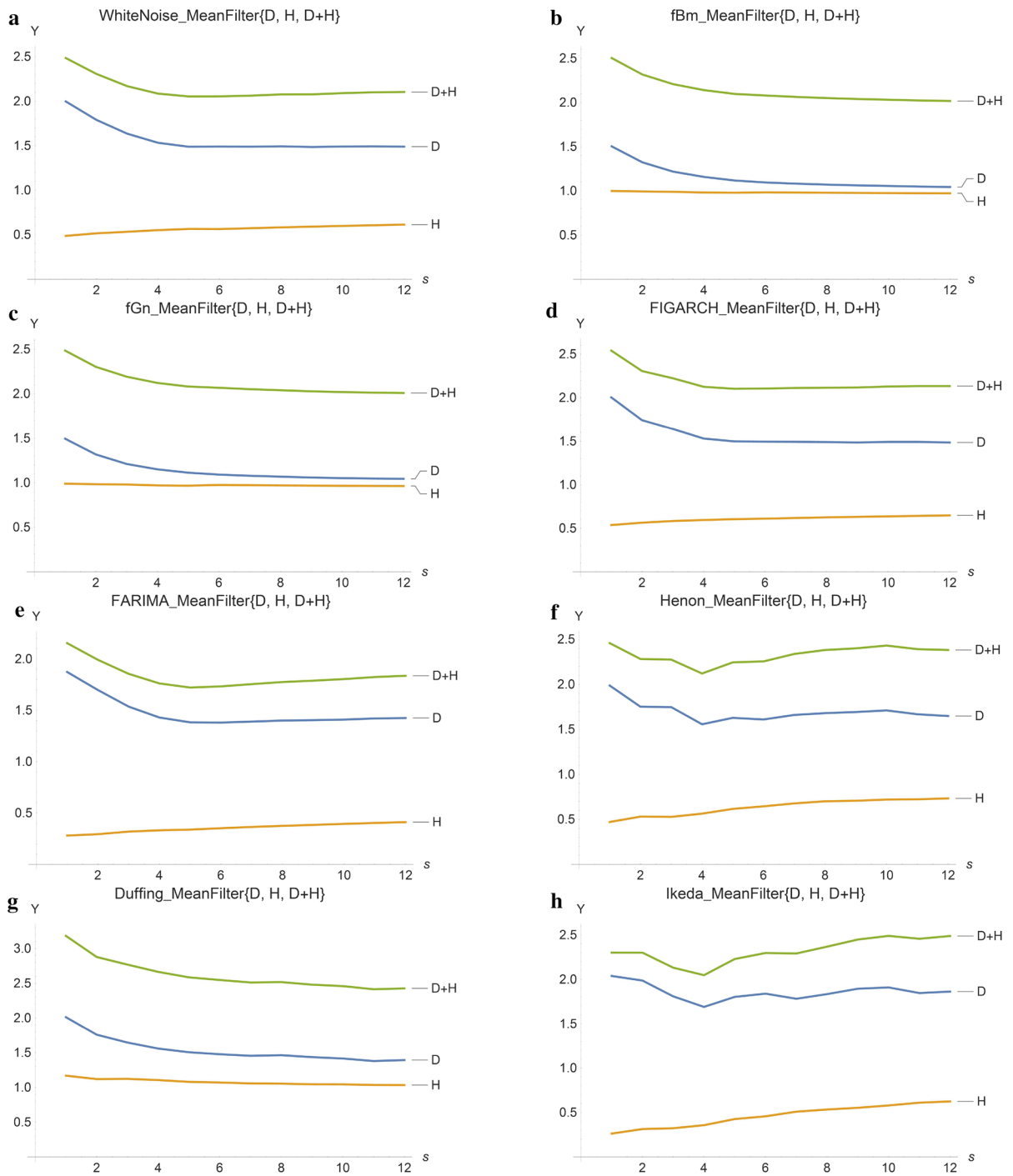
Fig. 6 Fractal dimension (D) and Hurst exponent (H) relationship in multiscale. **a** White noise, **b** fBm, **c** fGn, **d** FIGARCH, **e** FARIMA, **f** Henon, **g** Duffing, **h** Ikeda

However, the subfigures coming after *e*, displaying the results of MSHG and *H* algorithms for the chaotic time series, Henon, Duffing and Ikeda indicate irregular and diverging patterns from the relationship of $D + H = 2$ observed in the subfigures of the stochastic time series. While any firm convergence is not quite detected, around the number of 2.5 or between ranges of 2 and 2.5, the sum of *D* and *H* seems to fluctuate through until the last computed scale of 12. These such variant observations show that how distinguishing properties the stochastic time series and the chaotic time series possess in terms of *H*, *D* and the sum of *H* and *D* in multiple scales revealed by MSHG and *H* methods.

5 Conclusions

This study introduces the multiscale Higuchi's fractal dimension (MSHG) method as a new complexity measure which captures multiscale properties for time series by employing mean filter as the coarse-graining procedure as a scaling filter and Higuchi's fractal dimension algorithm as a self-similarity and complexity measurement method.

The choice of the most suitable scaling filter is experimented by applying a number of popular filtering methods on the stochastic and chaotic time series for which several filters are shown to be potentially effective as an addition to commonly used coarse-graining procedure in the multiscale literature. Consequently, the mean filter with an overlapping window is used for providing stable and effective results with the minimal data loss at each step. Also, Higuchi's fractal algorithm is employed because of its more accurate and faster execution even with the smaller data sets than the alternative algorithms for the computation of fractal dimension.

MSHG method is demonstrated on various selected stochastic time series and chaotic time series. Distinguishing results between these two different class of time series clearly are observed as supporting method's applicability and functionality as an alternative extension of current multiscale and monoscale complexity measuring methods.

Hurst exponent quantifies the persistence or long-range dependence of time series and has an infamous relationship of $D + H = 2$ with the self-similarity and complexity measure of fractal dimension. Further-

more, how this relationship between Hurst exponent and fractal dimension stands in multiple scales is examined by employing MSHG algorithm and stretching Hurst calculation with long-established Maldelbrot's method through multiple scales simultaneously, once again on the same time series data sets. While this relationship is observed to be holding for the stochastic time series, contrary evidence is emerged for the chaotic time series. The outcomes of these calculations, clearly, point out specific patterns for the stochastic time series and chaotic time series, through multiple scales with regard to the sum value of *D* and *H* which suggests the possible use of these unique multiscale features for the categorization of time series.

References

1. Mandelbrot, B.B.: The Fractal Geometry of Nature, vol. 173. WH Freeman, New York (1983)
2. Reishofer, G., Koschutnig, K., Enzinger, C., Ebner, F., Ahammer, H.: Fractal dimension and vessel complexity in patients with cerebral arteriovenous malformations. PLoS ONE **7**(7), e41148 (2012)
3. Mustafa, N., Ahearn, T.S., Waiter, G.D., Murray, A.D., Whalley, L.J., Staff, R.T.: Brain structural complexity and life course cognitive change. Neuroimage **61**(3), 694–701 (2012)
4. Acharya, R., Bhat, P.S., Kannathal, N., Rao, A., Lim, C.M.: Analysis of cardiac health using fractal dimension and wavelet transformation. ITBM-RBM **26**(2), 133–139 (2005)
5. Watari, S.: Fractal dimensions of solar activity. Solar Phys. **158**(2), 365–377 (1995)
6. Kalauzi, A., Cukic, M., Millán, H., Bonafoni, S., Biondi, R.: Comparison of fractal dimension oscillations and trends of rainfall data from Pastaza Province, Ecuador and Veneto, Italy. Atmos. Res. **93**(4), 673–679 (2009)
7. Lee, E.-T., Eun, H.-C.: Damage detection of damaged beam by constrained displacement curvature. J. Mech. Sci. Technol. **22**(6), 1111–1120 (2008)
8. Zhou, Z.-M.: Measurement of time-dependent fractal dimension for time series of silicon content in pig iron. Phys. A **376**, 133–138 (2007)
9. Klonowski, W., Olejarczyk, E., Stepien, R.: A new simple fractal method for nanomaterials science and nanosensors. Mater. Sci.-Poland **23**(3), 607–612 (2005)
10. Samadder, S., Ghosh, K., Basu, T.: Fractal analysis of prime indian stock market indices. Fractals **21**(01), 1350003 (2013)
11. Higuchi, T.: Approach to an irregular time series on the basis of the fractal theory. Phys. D **31**(2), 277–283 (1988)

12. Costa, M., Goldberger, A.L., Peng, C.-K.: Multiscale entropy analysis of biological signals. Phys. Rev. E **71**(2), 021906 (2005)
13. Li, D., Li, X., Liang, Z., Voss, L.J., Sleigh, J.W.: Multiscale permutation entropy analysis of eeg recordings during sevoflurane anesthesia. J. Neural Eng. **7**(4), 046010 (2010)
14. Ouyang, G., Dang, C., Li, X.: Complexity analysis of eeg data with multiscale permutation entropy. In: Advances in Cognitive Neurodynamics (II), pp. 741–745, Springer, Berlin (2011)
15. Morabito, F.C., Labate, D., La Foresta, F., Bramanti, A., Morabito, G., Palamara, I.: Multivariate multi-scale permutation entropy for complexity analysis of alzheimer's disease eeg. Entropy **14**(7), 1186–1202 (2012)
16. Shaobo, H., Kehui, S., Huihai, W.: Modified multiscale permutation entropy algorithm and its application for multiscroll chaotic systems. Complexity **21**(5), 52–58 (2016)
17. Wu, S.-D., Wu, C.-W., Lee, K.-Y., Lin, S.-G.: Modified multiscale entropy for short-term time series analysis. Phys. A **392**(23), 5865–5873 (2013)
18. Azami, H., Escudero, J.: Improved multiscale permutation entropy for biomedical signal analysis: interpretation and application to electroencephalogram recordings. Biomed. Signal Process. Control **23**, 28–41 (2016)
19. Zhao, X., Sun, Y., Li, X., Shang, P.: Multiscale transfer entropy: measuring information transfer on multiple time scales. Commun. Nonlinear Sci. Numer. Simul. **62**, 202–212 (2018)
20. Nixon, M., Aguado, A.S.: Feature Extraction and Image Processing for Computer Vision. Academic Press, New York (2012)
21. Vaseghi, S.V.: Advanced Digital Signal Processing and Noise Reduction. Wiley, New York (2008)
22. Fukunaga, K., Hostetler, L.: The estimation of the gradient of a density function, with applications in pattern recognition. IEEE Trans. Inf. Theory **21**(1), 32–40 (1975)
23. Comaniciu, D., Meer, P.: Mean shift: a robust approach toward feature space analysis. IEEE Trans. Pattern Anal. Mach. Intell. **5**, 603–619 (2002)
24. Paris, S., Kornprobst, P., Tumblin, J., Durand, F.: A gentle introduction to bilateral filtering and its applications. In: ACM SIGGRAPH 2007 Courses, p. 1. ACM (2007)
25. Tomasi, C., Manduchi, R.: Bilateral filtering for gray and color images. In: ICCV, vol. 98, p. 2 (1998)
26. Rudin, L.I., Osher, S., Fatemi, E.: Nonlinear total variation based noise removal algorithms. Phys. D **60**(1–4), 259–268 (1992)
27. Selesnick, I.W., Bayram, I.: Total Variation Filtering. White paper (2010)
28. Standard deviation filter. https://reference.wolfram.com/language/ref/StandardDeviationFilter.html. Accessed 11 Sept 2019
29. Burger, W., Burge, M.J.: Digital Image Processing: An Algorithmic Introduction Using Java. Springer, Berlin (2016)
30. Marques, O.: Practical Image and Video Processing Using MATLAB. Wiley, New York (2011)
31. Harmonic mean filter. https://reference.wolfram.com/language/ref/HarmonicMeanFilter.html. Accessed 11 Sept 2019
32. Gao, J., Cao, Y., Tung, W.-W., Hu, J.: Multiscale Analysis of Complex Time Series: Integration of Chaos and Random Fractal Theory, and Beyond. Wiley, New York (2007)
33. Mandelbrot, B.B.: Fractals and Scaling in Finance: Discontinuity, Concentration, Risk. Selecta Volume E. Springer, Berlin (2013)
34. Fernández-Martínez, M., Sánchez-Granero, M.: Fractal Dimension for Fractal Structures with Applications to Finance. Springer, Switzerland (2019)
35. Coyt, G.G., Diosdado, A.M., Brown, F.A., et al.: Higuchi's method applied to the detection of periodic components in time series and its application to seismograms. Rev. Mex. de Física S **59**(1), 1–6 (2013)
36. Arqub, O.A., Shawagfeh, N.: Application of reproducing kernel algorithm for solving dirichlet time-fractional diffusion-gordon types equations in porous media. J. Porous Media **22**(4), 411–434 (2019)
37. Accardo, A., Affinito, M., Carrozzi, M., Bouquet, F.: Use of the fractal dimension for the analysis of electroencephalographic time series. Biol. Cybern. **77**(5), 339–350 (1997)
38. Mandelbrot, B.B., Van Ness, J.W.: Fractional brownian motions, fractional noises and applications. SIAM Rev. **10**(4), 422–437 (1968)
39. Brooks, C.: Introductory Econometrics for Finance. Cambridge University Press, Cambridge (2008)
40. Box, G.E., Jenkins, G.M., Reinsel, G.C., Ljung, G.M.: Time Series Analysis: Forecasting and Control. Wiley, New York (2015)
41. Sheng, H., Chen, Y., Qiu, T.: Fractional Processes and Fractional-Order Signal Processing: Techniques and Applications. Springer, Berlin (2011)
42. Granger, C.W., Joyeux, R.: An introduction to long-memory time series models and fractional differencing. J. Time Ser. Anal. **1**(1), 15–29 (1980)
43. Baillie, R.T., Bollerslev, T., Mikkelsen, H.O.: Fractionally integrated generalized autoregressive conditional heteroskedasticity. J. Econom. **74**(1), 3–30 (1996)
44. Yilmaz, A., Unal, G.: Chaoticity properties of fractionally integrated generalized autoregressive conditional heteroskedastic processes. Bull. Math. Sci. Appl. **15**, 69–82 (2016)
45. Hénon, M.: A two-dimensional mapping with a strange attractor. In: The Theory of Chaotic Attractors, pp. 94–102. Springer, Berlin (1976)
46. Wiggins, S.: Introduction to Applied Nonlinear Dynamical Systems and Chaos, vol. 2. Springer, Berlin (2003)
47. Ikeda, K.: Multiple-valued stationary state and its instability of the trasmitted light by a ring cavity system. Opt. Commun. **30**(2), 257–261 (1979)
48. Tong, S., Zhang, J., Bao, Y., Lai, Q., Lian, X., Li, N., Bao, Y.: Analyzing vegetation dynamic trend on the mongolian plateau based on the hurst exponent and influencing factors from 1982–2013. J. Geograph. Sci. **28**(5), 595–610 (2018)
49. Ohu, I.P., Carlson, J.N., Piovesan, D.: The hurst exponent: a novel approach for assessing focus during trauma resuscitation. In: Signal Processing in Medicine and Biology, pp. 139–160. Springer, Berlin (2020)
50. Singh, A., Bhargawa, A.: An early prediction of 25th solar cycle using hurst exponent. Astrophys. Space Sci. **362**(11), 199 (2017)

51. Ramos-Requena, J.P., Trinidad-Segovia, J., Sánchez-Granero, M.: Introducing hurst exponent in pair trading. Phys. A **488**, 39–45 (2017)

52. Hurst, H.E.: Long-term storage capacity of reservoirs. Trans. Am. Soc. Civ. Eng. **116**, 770–799 (1951)

53. Pipiras, V., Taqqu, M.S.: Long-Range Dependence and Self-similarity, vol. 45. Cambridge University Press, Cambridge (2017)

54. Mandelbrot, B.: Long-run linearity, locally gaussian process, h-spectra and infinite variances. Int. Econ. Rev. **10**(1), 82–111 (1969)

55. Raimundo, M.S., Okamoto Jr., J.: Application of hurst exponent (h) and the r/s analysis in the classification of forex securities. Int. J. Model. Optim **8**, 116–124 (2018)