Proceedings of the 8th World Congress on Electrical Engineering and Computer Systems and Sciences (EECSS'22)
Prague, Czech Republic - July 28- 30, 2022
Paper No. MVML 109
DOI: 10.11159/mvml22.109

Precision and Accuracy of Length and Variance Fractal Dimensions Computed from Fractional Self-Affine Signals

Soleiman Hosseinpour[1], Witold Kinsner[2], and **Nariman Sepehri[1]**
[1] Department of Mechanical Engineering, University of Manitoba, Winnipeg, MB, Canada
[2] Department of Electrical and Computer Engineering, University of Manitoba, Winnipeg, MB, Canada
Email: hossein8@myumanitoba.ca, Witold.Kinsner@umanitoba.ca, Nariman.Sepehri@umanitoba.ca.

***Abstract* -** Many digital signal processing algorithms are based on mono-scale analysis. Since the reshuffling of any data value does not change the probability distribution, the sense of correlation or covariance in the data is lost, especially when dealing with self-affine time series. An alternative approach is to use poly-scale analysis. This paper describes two poly-scale algorithms: the length fractal dimension and variance fractal dimension and evaluates their accuracy and precision. First, we generate white Gaussian noise and fractal Brownian motion using the concepts of fractional Brownian motion and the discrete Fourier transform. Next, we evaluate the performance of these measures. Since many processes are nonstationary, we also present a stationarity-frame detection technique based on the One-Way ANOVA and Bartlett tests. Our results show that these poly-scale techniques can always be used as reliable tools for different purposes in self-affine data analysis.

***Keywords*:** Signal processing; poly-scale analysis; variance fractal dimension; length fractal dimension; stationarity frames.

1. Introduction

Methods for digital signal processing (DSP) have evolved significantly in recent years due to the diverse applications of DSP in areas such as data transmission, speech and image recognition/translation, and biomedical applications [1]. Many of these methods treat the digital signals at a single-scale or mono-scale. Recently, multi-scale and poly-scale analyses have been gaining attention because they provide more insight into the internal structure of the data. However, to achieve good results, these emerging analyses require much attention to detail. One of the most applicable poly-scale methods is based on the concepts of single and multiple power laws in the form of complexity measures, including fractal dimensions, as many physical processes have the characteristics of fractals [2]. Fractals are complex objects that some parts of them replicate the structure of the whole. The typical examples of fractals are the Koch curve and snowflake, Sierpinski triangle, and Fibonacci spiral (self-similarity at point of convergence). Such self-similar, self-affine objects require moving from the integer-dimensional space to a fractional space. In recent years, this thought process has had significant advantages from both theoretical and practical points of view.

Variance and length fractal dimensions are two important poly-scale measures used in many data analysis areas [3] to [9]. These poly-scale methods and related algorithms were introduced and developed by Kinsner in 1994 [4]. Unlike a mono-scale analysis, a poly-scale analysis measures a signal at multiple scales (covers with volume elements, “vels” for short). It considers the measures from each scale cover simultaneously. If the time series is non-stationary, the analysis can be done within a moving stationarity frame to form a complexity measure trajectory. For example, the variance fractal dimension (VFD), D_σ, has been used in applications such as nonstationary speech segmentation [5]. Multiscale and poly-scale analysis and synthesis using VFD in cognitive systems were introduced in [6]. Feature extraction of DNA sequences based on multifractal analysis was performed in [7]. Another important advantage of VFD analysis is that it is not sensitive to noise and can be applied in real-time streaming data [3]. Other applications include the detection of anomalies in (i) traffic analysis, specifically in the distributed denial of service (DDOS) attacks, and (ii) nonlinear hydraulic actuators used in heavy-duty machinery.

This paper presents such a poly-scale analysis and synthesis, emphasizing the accuracy and precision of the length and variance fractal dimensions, D_L and D_σ, respectively. First, we study the transient duration of the random number generator to be sure that the generator transient does not affect our data. Next, we investigate two methods to synthesize two classes of fractional noise with known fractal properties. We further describe two statistical tests, the One-Way ANOVA and Bartlett tests, to determine the appropriate stationarity frame of the synthesized time series. We evaluate the performance of the proposed complexity measures with 35 different vel sizes, using the technique described in [6], [8] and [9].

2. Fractional Noise Synthesis

Two classes of time series are synthesized for the required study: (i) a white uncorrelated Gaussian noise with the stochastic self-similarity fractal dimension $D_{SS} = 2$, and (ii) a Brownian noise with $D_{SS} = 1.5$. The time series were generated using the following two methods: (a) the fast Fourier Transform (FFT) and (b) the covariance generator.

The FFT allows us to generate fractional noise through the concept of spectral exponent β in the frequency domain. One of the standard conventions is to present the spectral slope of the noise as directly proportional to the power of frequency. For a self-affine time series, the power spectral density (PSD) satisfies the following power-law relation

$$PSD \sim \frac{1}{f^{\beta}} \tag{1}$$

It is more convenient to work with the amplitude spectrum density (ASD). Based on the well-known relation that ASD is the square root of the PSD, we manipulate the ASD instead of PSD to obtain the desired PSD slope. First, we need to synthesize Gaussian noise in the time domain and get the fast Fourier coefficient as [10]

$$Y[k] = \frac{1}{N}\sum_{n=1}^{N} y[n]e^{-j2\pi\frac{nk}{N}} \tag{2}$$

where $Y[k]$ is the kth fast Fourier coefficient of Gaussian noise, $y[n]$ is the signal in the time domain, and N is the frame size. To generate the Brownian noise, we can take the first-order integral of the Gaussian noise, which is equivalent to $k^{-1}Y[k]$ in the frequency domain. Now, we can generate the Brownian noise by taking the inverse fast Fourier transform as follows

$$b[k] = \sum_{n=1}^{N} (k^{-1}Y[k])e^{j2\pi\frac{nk}{N}} \tag{3}$$

where $b[k]$ is the Brownian noise in the time domain. The fractional Brownian motion generator involves a continuous zero-mean Gaussian process $\{W_t, t \geq 0\}$ with covariance function [11]

$$cov(W_t, W_s) = \frac{1}{2}(|t|^{2H} + |s|^{2H} - |t-s|^{2H}), t, s \geq 0 \tag{4}$$

is called fractional Brownian motion (FBM) with Hurst parameter $H \in (0,1)$. It can be shown that in self-affine time series, the variance is related to the time increments according to the following power-law relation as

$$Var[y(t) - y(s)] \sim |t-s|^{2H} \tag{5}$$

where y is a discrete sample of a time series sampled at time t. Taking the logarithm of both sides and simplifying allows us to extract the Hurst exponent

$$H = \frac{1}{2}\lim_{\triangle t \to 0} \frac{\log[Var[\triangle y]]}{\log[\triangle t]} \tag{6}$$

Evaluating the variance fractal dimension based on the Hurst exponent can be carried out using the following expression

$$D_{\sigma} = E + 1 - H \tag{7}$$

For a time series with a single independent variable $E = 1$ and $D_{\sigma} \in [1,2]$. Note that $H = 0.5$ corresponds to the Brownian motion whose $D_{\sigma} = 1.5$.

To perform a poly-scale analysis, we consider the displacement $\triangle t = |t-s|$ at multiple scales and different frames produced by the time series $y[N_E]$.

Notice that we have selected this approach to generate fractional noise because it can produce signals of any complexity such as $D_{\sigma} = 1.75$ and $D_{\sigma} = 1.25$ using Eq. (4) and with $H = 0.25$ and $H = 0.75$, respectively. Since only one algorithm is needed to produce the noise, the differences originating from different algorithm implementations are eliminated.

3. Study of Generator Transients in a Short Time Series

In this section, we synthesize a strictly stationary Gaussian stochastic process (white noise) called TSw128k in the form of a time series $y[N_E]$ where N_E is the number of samples in the entire series. To generate the white Gaussian noise, we consider the Mersenne Twister generator with a seed of 1. The number of samples is $N_E = 2^{12} = 4096$ (or 4K) with

$G(\mu, \sigma) = G(0,1)$. In general, transients may occur on some generators, particularly those used in chaotic dynamical systems. Since transients are usually not discussed in various language implementations and numerical applications, very accurate
experiments should address the problem by skipping the first N_T samples. In order to determine if the generator has a transient, and how long it lasts, we generate the time series after skipping the following transient sequences

- $N_{T1} = 0$ (skip no samples);
- $N_{T2} = 128$ (skip the first 128 samples);
- $N_{T3} = 256$ (skip the first 256 samples);
- $N_{T4} = 512$ (skip the first 512 samples);
- $N_{T5} = 1024$ (skip the first 1024 samples).

Now, we plot the first $N_P = 512$ samples from each instance of the time series with N_{T1} to N_{T5} in Fig. 1. To achieve a better visualization, the amplitudes in samples are shifted by five units after N_{T1}. As we can see, there is no transient in this pseudorandom data generator. Then the used generator skips the transient samples automatically. Table 1 compares some descriptive statistical properties for all the entire Gaussian test sequences $TSw4K_1$ to $TSw4K_5$.

Table 1: Statistical properties for $TSw4K_1$ to $TSw4K_5$.

Statistical Moment	$TSw4K_1$	$TSw4K_2$	$TSw4K_3$	$TSw4K_4$	$TSw4K_5$	Normalized Range
Maximum	3.4562	3.4562	3.4562	3.4562	3.4562	0
Minimum	-3.0652	-3.0652	-3.1234	-3.1234	-3.1234	0.0186
Range	6.5214	6.5214	6.5796	6.5796	6.5796	0.0186
Sum	34.1948	35.4395	-2.9251	8.2154	55.2310	1.0530
Mean	0.0083	0.0087	-0.0007	0.0020	0.0135	1.0519
Median	0.0226	0.0193	0.0026	0.0001	0.0211	0.9956
Mode	-3.0652	-3.0652	-3.1234	-3.1234	-3.1234	0.0186
Sample Variance	1.0224	1.0216	1.0275	1.0218	1.0184	0.0089
Standard Deviation	1.0111	1.0107	1.0136	1.0109	1.0092	0.0043
Skewness	0.0214	0.0197	0.0301	0.0217	0.0229	0.3455
Kurtosis	2.9935	2.9985	3.0016	2.9985	2.9368	0.0216
Standard Error	2.4686×10^{-4}	2.4676×10^{-4}	2.4747×10^{-4}	2.4679×10^{-4}	2.4638×10^{-4}	0.0044

We define the normalized range for variation of each descriptive statistics $TSw4K_1$ to $TSw4K_5$ as

$$\text{Normalized Range} = \frac{\max(TSw4K_i) - \min(TSw4K_i)}{\max(TSw4K_i)} \tag{8}$$

where $i = 1,2,3,4,5$. The maximum values for all $TSw4K_i$ are the same, while minimum values slightly changed. Based on Eq. (8) the values of sum, mean, and median have changed more than other statistics.

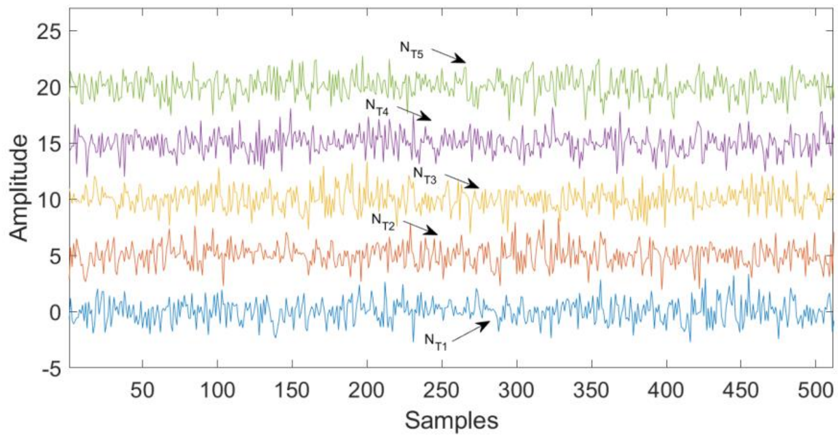

Fig. 1. The first $N_P = 512$ samples N_{T1} to N_{T5}.

4. Accuracy and Precision of Complexity Measures

The white Gaussian noise can be generated in many ways. First, we can generate it using pseudorandom generators in Matlab with $G(\mu, \sigma) = G(0,1)$. Another method is based on Eq. (4) with $H \to 0$. Here, we generate a white Gaussian noise called TSw128K with $N_E = 2^{17}$ (or $128K = 131072_{10}$) using the Mersenne Twister generator with a seed of 1. Before studying the accuracy and precision of the length fractal dimension, we explain the stationarity method that we used.

4.1. Stationarity

To have a valid analysis, either the entire signal recording (epoch), N_E, should be stationary, or we should find a frame size in which that signal is stationary. Signal processing often requires weak-sense stationarity (WSS). A stochastic process $Y_t: t \in T$, is WSS if mean and autocovariance are constant and do not change over time. In other words

$$E[Y_t] = E[Y_{t+h}] \tag{9}$$

$$\gamma(h) = Cov(Y_t, Y_{t+h}),\ h \in T \ such\ that\ t + h \in T \tag{10}$$

the second condition implies that the autocovariance is independent of $t \in T$. Note that this also shows the variance of the process is constant over time.

To check for the stationarity, there are several methods that we can use such as the Augmented Dickey-Fuller test, Haar wavelet-based variance change, and Kwiatkowski-Phillips Schmidt-Shin test. However, in this work, we utilize the One-Way ANOVA test for stationarity in mean and Bartlett's test to establish the stationarity for variance.

The One-Way ANOVA parametric test is a tool to check whether there is statistical evidence that the associated population means are significantly different in two or more independent groups. This test computes the ratio of between-group variation to within-group variation. We denote by SSR the variation between groups as

$$SSR = \sum_{j=1}^{k} n_j (\bar{y}_j - \bar{y})^2 \tag{11}$$

where $\bar{y}_j$ is the sample mean of the group j, and $\bar{y}$ is the overall mean of the sample. The variation within groups or SSE is defined as

$$SSE = \sum_{i=1}^{N} \sum_{j=1}^{k} (y_{ij} - \bar{y}_j)^2 \tag{12}$$

We also use SST to indicate the total sum of squares as

$$SST = \sum_{i=1}^{N} \sum_{j=1}^{k} (y_{ij} - \bar{y})^2 \tag{13}$$

then we have the following relation

$$SST = SSR + SSE \tag{14}$$

Now, the test statistic is defined as

$$F = \frac{(N-k)SSR}{(k-1)SSE} \tag{15}$$

where k is the number of groups, and N is the total number of observations. If this value is significantly higher than the critical value in the table of F-distribution with $(k-1, N-k)$ degrees of freedom and significance levels 0.05, then you can reject the null hypothesis that the means among groups are significantly different from each other.
Bartlett's test is also used to determine if multiple samples are from populations with equal variances. The test statistic is

$$B = \frac{(N-k)\ln(\sigma_p^2)-\sum_{i=1}^{k}(N_i-1)\ln(\sigma_i^2)}{1+(1/(3(k-1)))((\sum_{i=1}^{k}1/(N_i-1))-1/(N-k)} \tag{16}$$

where σ_i^2 is the variance of the $i$the group, N is the total sample size, N_i is the sample size of the $i$the group, k is the number of groups, and σ_p^2 is the pooled variance. The pooled variance is defined as

$$\sigma_p^2 = \sum_{i=1}^{N}\frac{(N_i-1)s_i^2}{(N-k)} \tag{17}$$

The test statistic has a chi-square distribution with $k-1$ degrees of freedom. If the p-value for the test is smaller than the significance level 0.05, then the test rejects the null hypothesis that all group means are equal and concludes that at least one of the group variances is different from the others. In our investigation, the frames of the whole epoch are considered as the independent populations that should be evaluated using these two statistical tests.

4.2. Length Fractal Dimension for White Gaussian Noise

We calculate the length fractal dimension, D_L, from the TSw128K using the batch processing described in [8]. First, using the method described in subsection 4.1, we show that the TSw128K is stationary. For the frame size $N_F = 512$, the p-values for the mean is 0.6889 and for variance is 0.2263. Since p-values are more than 0.05, they indicate that One-Way ANOVA and Bartlett tests do not reject the null hypothesis that the mean and variance are equal. Then the frame sizes are stationary.

Now, we describe the process of computing the length fractal dimension, D_L. The following covers have been considered, with volume elements (vels) of unequal sizes in order to generate a large number of measures with a relatively equal distribution in the log-log values

$$n_k = \lfloor 2^{\frac{k+5}{4}} \rfloor, k = 1,2,\cdots,35 \tag{18}$$

where $\lfloor\cdot\rfloor$, is the floor function. It is seen that the above 35 vel sizes cover the TSw128K quite well. The last vel size is 1024 samples apart and covers the TSW128K 128 times. This is four times more than the statistical 32 minimum number of covers. For each index $k = 1$ to 35 and the corresponding vel size n_k, we compute the length of each subsequence as

$$(L_{k,m})_j = |y[m+jn_k] - y[m+(j-1)n_k]| \tag{19}$$

where m is the subsequence interval, j is the element in the subsequence, and k is the subsequence length. The total for each subsequence can be taken as follows

$$L_{k,m} = \sum_{j=1}^{J_k} |y[m+jn_k] - y[m+(j-1)n_k]| \tag{20}$$

where J_k is the number of vels of size n_k in the subsequences of differences and is defined as

$$J_k = \lfloor \frac{N_F - n_k}{n_k} \rfloor \tag{21}$$

and $\lfloor\cdot\rfloor$ is again the floor function and N_F is the stationary frame size. Here, we consider the $N_F = N_E$, to obtain length the fractal dimension D_L for the time series TSw128K. Then, we must take the average length, $L_{k,ave}$ from all the subsequences as follows

$$L_{k,ave} = \frac{1}{n_k}\sum_{m=1}^{n_k} L_{k,m} \tag{22}$$

We calculate the logarithms of the $L_{k,ave}$ measure for the point on the doubly-logarithmic or log-log plot [x-axis, y-axis]

$$(\log_2(n_k), \log_2(L_{k,ave})) \tag{23}$$

For all the 35 points, we fit a line, using a robust linear regression method called iteratively reweighted least squares (IRLS) [12]. Next, we calculate the slope of the line. Then, the fractal dimension for the TSw128K process is as follows

$$D_L = 1 - s_\triangle \tag{24}$$

where $s_\triangle$ is the slope of the line in the log-log plot [x-axis, y-axis]. The process is repeated 100 times and resulted in

$D_L = 2.0008$ with the standard deviation $\sigma = 1.6906 \times 10^{-4}$. Figure 2 shows the fitted line for the 35 points in the log-log plot.

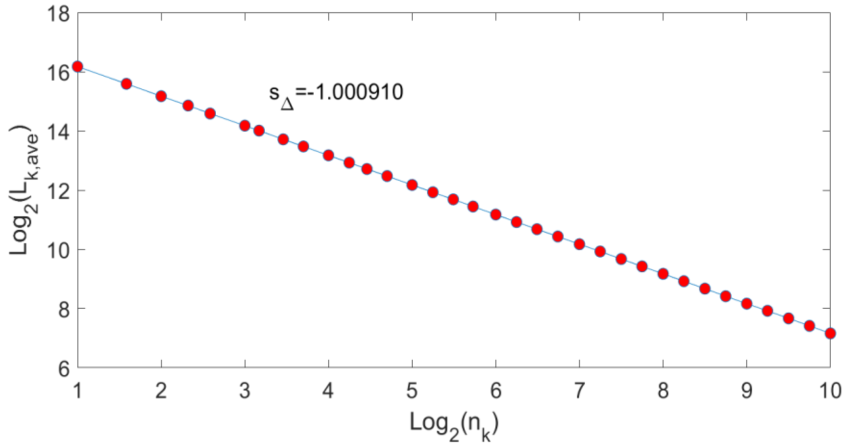

Fig. 2. The log–log plot for calculating the length dimension, D_L, of TSw128K.

4.3. Length and Variance Fractal Dimension of Brownian Noise

Let us focus on the Brownian self-affine time series called TSb128K, whose statistical self-similarity fractal dimension is $D_\sigma = 1.5$. It is another simplest form of a self-affine signal which is suitable for studying the accuracy and precision of the poly-scale complexity measures D_L and D_σ. Here, we synthesize the time series TSb128K with $N_E = 2^{17}$ (or $128K = 131072_{10}$) samples. To do this, we use the following generator

$$y_0 = 0 \quad (25)$$

$$y_{j+1} = y_j + g(j), \text{for } j = 1,2,\cdots,N_E \quad (26)$$

where $g(j)$ is the Gaussian process with $G(0,1)$ as discussed in previous sections. Note that we skip $1K$ samples to minimize any possible impact of a transient. Figure 3 shows the generated Brownian noise using recursive sequence (25)-(26). We used the method described in Subsection 4.2 to calculate the length fractal dimension for Brownian noise TSb128K. We performed the algorithm 100 times and achieved the length fractal dimension average $D_L = 1.5006$ with standard deviation $\sigma = 0.0059$ for these iterations. Figure 4 shows the fitted line for the 35 points in the log-log plot.

We can also define the variance fractal dimension, D_σ, using the method described in [9]. We compute the variance of each subsequence $V_{k,m}$ for the corresponding vel size n_k in Eq. (18) as

$$V_{k,m} = \frac{1}{J_k - 1}\left[\sum_{j=1}^{J_k} (y[m + jn_k] - y[m + (j-1)n_k])^2 - \frac{1}{J_k}\left(\sum_{j=1}^{J_k} y[m + jn_k] - y[m + (j-1)n_k]\right)^2\right] \quad (27)$$

where $J_k > 1$, and $m = 1,2,\cdots,n_k \in \mathbb{N}$. If we take the average of the subsequences as

$$V_{k,ave} = \frac{1}{n_k}\sum_{m=1}^{n_k} V_{k,m} \quad (28)$$

for each scale n_k we have the following pairs

$$(\log_2(n_k), \log_2(V_{k,ave})) \quad (29)$$

Now, we use IRLS as a robust linear regression to compute the slope of the line in each frame. Then, the variance fractal dimension can be obtained as follows

$$D_\sigma = 2 - H = 2 - \frac{1}{2}s_\triangle \quad (30)$$

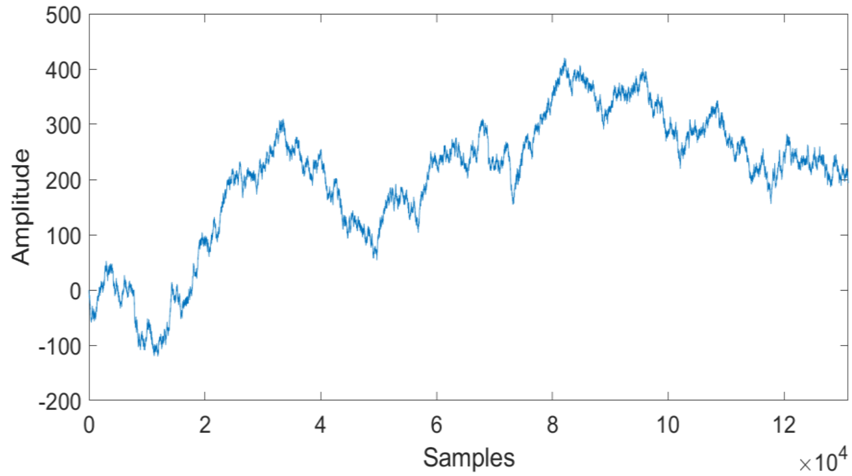

Fig. 3. Brownian self-affine time series TSb128K with $N_E = 2^{17}$ samples.

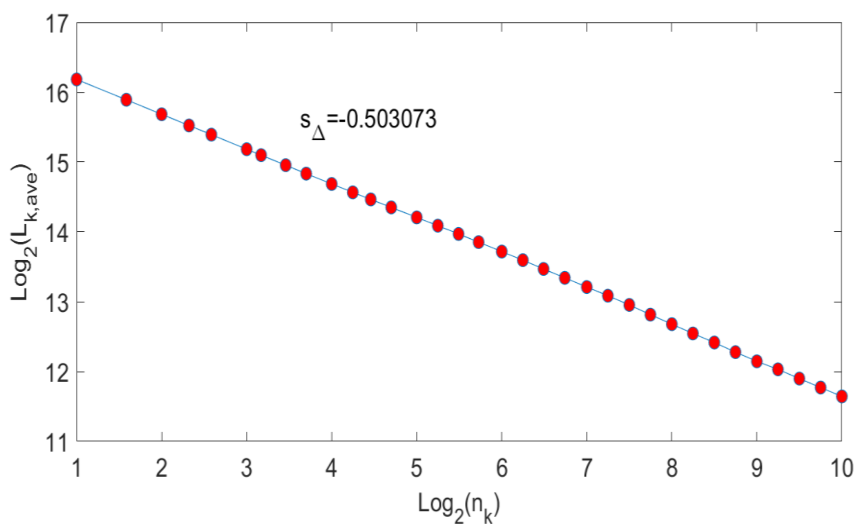

Fig. 4. The log–log plot for calculating the length fractal dimension, D_L, of TSb128K.

The average value of the variance fractal dimension after 100 times iteration is $D_\sigma = 1.5001$ with standard deviation $\sigma = 0.0066$. In Fig. 5, we can see the obtained fitted line using IRLS for the 35 points in the log-log plot. It must be stressed that in the calculation of variance and length fractal dimension we should eliminate outliers and saturation points from the log-log plot values. The IRLS can mitigate the influence of outliers, otherwise, errors might be significant. In addition, we can only apply a linear regression when the pairs form a line. If the points do not form a line, the power-law may vary over the range considered, thus implying a multifractal. The log-log plots for calculating variance and length fractal dimension in Figs. 4 and 5 show that the 35 points form a straight line as we expected.

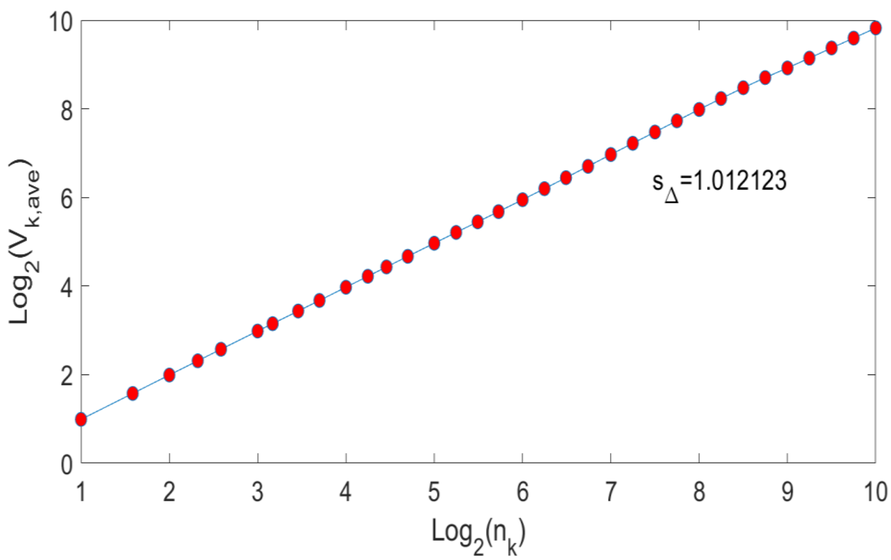

Fig. 5. The log–log plot for calculating the variance fractal dimension, D_σ, of TSb128K.

5. Conclusion

This paper presents an analysis of the accuracy and precision of two classes of complexity measures called length fractal dimension, D_L, and variance fractal dimension, D_σ. Since the fractal complexity of the white Gaussian and Brownian noises are known, the two processes are synthesized as time series called TSw128K and TSb128K, respectively, to investigate the performance of these D_L and D_σ algorithms. While there are many techniques suitable to synthesize fractional self-affine noise with a specified complexity measure, the discrete-Fourier transform is a technique that could be advantageous from the practical and theoretical points of view for generating different classes of fractional noise for testing and experimentation. This single technique of synthesizing a fractional noise can be used not only to verify the accuracy of poly-scale measures but also to evaluate signals contaminated by a known fractional noise.

Stationarity is one of the critical issues in signal processing. Thus, this paper further applies a method based on the One-Way ANONA and Bartlett tests that evaluate whether the variance and means of two or multiple samples are noticeably different. Using this technique one can identify an appropriate stationary frame size, even when the entire time series is not stationary, and can perform the poly-scale analysis of data of interest within the stationarity frame.

References

[1] Zahir M. Hussain, Amin Z. Sadik, and Peter O'Shea, *Digital Signal Processing, An Introduction with MATLAB and Applications*. New York, NY: Springer, 2011, 371 pages {ISBN-13: 978-3642155901}.

[2] Witold Kinsner, "*A robust poly-scale length complexity measure for stochastic self-affine processes*," *in Proc. IEEE Intern. Conf. Cognitive Informatics and Cognitive Computing, ICCI*CC'21*, Paper 83 (Banff, AB; Oct 29-30, 2021), 2021.

[3] Witold Kinsner, "*Batch and real-time computation of a fractal dimension based on variance of a time series*," *Technical Report*, DEL94-6, Dept. Electrical & Computer Eng., University of Manitoba, June 1994.

[4] Witold Kinsner, "Fractal dimensions: Morphological, entropy, spectrum, and variance classes," Technical Report. Winnipeg, MB: University of Manitoba, 1994.

[5] Warren Grieder, and Witold Kinsner, "*Speech segmentation by variance fractal dimension,*" *in Proc. Canadian Conference on Electrical and Computer Engineering* (Halifax, NS, Canada), pp. 481-485, Sept 1994.{DOI:

10.1109/CCECE.1994.405793 }.

[6] Witold Kinsner, “*It’s time for multiscale analysis and synthesis in cognitive systems*," *in Proc. IEEE 10th International Conference on Cognitive Informatics and Cognitive Computing*, ICCI-CC'11 (Banff, AB, Canada), pp. 7-10, Aug 2011. {DOI: 10.1109/COGINF.2011.6016116 }.

[7] Witold Kinsner, and Hong Zhang, “*Multifractal analysis and feature extraction of DNA sequences*," *in Proc. 8th IEEE International Conference on Cognitive Informatics (Hong Kong, China)*, pp. 29-37, June 2009. {DOI: 10.1109/COGINF.2009.5250696 }.

[8] Witold Kinsner, *Fractal and Chaos Engineering: Mono-scale, Multiscale and Poly-scale Analyses.* Winnipeg, MB: OCO Research, February 2020, 1106 pages. {ISBN: 978-0-9939347-2-8, eBook}.

[9] Witold Kinsner, "*A robust variance complexity measure for stochastic self-affine processes*," *in Proc. 18th IEEE International Conference on Cognitive Informatics & Cognitive Computing, ICCI*CC19 (Polytechnic University of Milan, Milan, Italy; July 23-25, 2019)*, pp. 75-82, Jul 2019. {DOI: 10.1109/ICCICC46617.2019.9146065}.

[10] Hristo Zhivomirov, “*A method for colored noise generation*.", *Romanian Journal of Acoustics and Vibration*, vol. 15, no. 1, pp. 14-19, 2018.

[11] Volker Schmidt, *Stochastic Geometry, Spatial Statistics and Random Fields, Models and Algorithms*. Springer, 2015, 488 pages, {ISBN-13: 978-3319100647}.

[12] Paul W. Holland, and Roy.E. Welsch, “*Robust regression using iteratively reweighted least-squares*,” *Communications in Statistics: Theory and Methods*, vol. 6, no. 1, pp. 13-827, 1977.