

# MFDFA: Efficient multifractal detrended fluctuation analysis in python<sup>☆</sup>

Leonardo Rydin Gorjão<sup>a,b,\*</sup>, Galib Hassan<sup>a,b</sup>, Jürgen Kurths<sup>c,d,e</sup>, Dirk Witthaut<sup>a,b</sup>

<sup>a</sup> Forschungszentrum Jülich, Institute for Energy and Climate Research - Systems Analysis and Technology Evaluation (IEK-S TE), 52428 Jülich, Germany

<sup>b</sup> Institute for Theoretical Physics, University of Cologne, 50937 Köln, Germany

<sup>c</sup> Potsdam Institute for Climate Impact Research, 14473 Potsdam, Germany

<sup>d</sup> Institute of Physics, Humboldt University Berlin, 12489 Berlin, Germany

<sup>e</sup> Centre for Analysis of Complex Systems, World-Class Research Center "Digital biodesign and personalised healthcare", Sechenov First Moscow State Medical University, 119991 Moscow, Russia

## ARTICLE INFO

### Article history:

Received 31 May 2021

Received in revised form 29 November 2021

Accepted 3 December 2021

Available online 8 December 2021

### Keywords:

Multifractal detrended fluctuation analysis

Time series analysis

Hurst coefficient

Multifractal spectrum

Singularity strength

## ABSTRACT

Multifractal detrended fluctuation analysis (MFDFA) has become a central method to characterise the variability and uncertainty in empiric time series. Extracting the fluctuations on different temporal scales allows quantifying the strength and correlations in the underlying stochastic properties, their scaling behaviour, as well as the level of fractality. Several extensions to the fundamental method have been developed over the years, vastly enhancing the applicability of MFDFA, e.g. empirical mode decomposition for the study of long-range correlations and persistence. In this article we introduce an efficient, easy-to-use python library for MFDFA, incorporating the most common extensions and harnessing the most of multi-threaded processing for very fast calculations.

© 2021 Elsevier B.V. All rights reserved.

## 1. Introduction

A common tool to unveil the nature of the scaling and fractionality of a process, natural or computer-generated, is Multifractal Detrended Fluctuation Analysis (MFDFA). It was initially developed by Peng et al. [1,2] as basic Detrended Fluctuation Analysis (DFA) and later extended to study multifractal processes by Kandelhardt et al., giving rise to MFDFA [3]. It addresses the question of the presence of correlations in time series and can be employed to analyse both discrete as well as continuous-time stochastic processes. Since its initial development in the late 90's, it has been revisited to incorporate several other elements, e.g. empirical mode decomposition as a method for detrending [4–7], overlapping moving windows [8,9], and a new metric denoted extended detrended fluctuation analysis [10–13]. There are several additional features exist, designed to study correlations of two or more time series [14,15], lag correlations in time series [16], and Fourier-DFA [17], amongst others. A comprehensive study of DFA and the interplay between trends in data and

correlated noise can be found in Ref. [18]. Multifractal time series analysis has found application in various fields, such as the analysis of heartbeat rate [19], arterial pressure [10], EEG sleep data [11,13], physiology [20], stroke time series from Parkinson's disease patients [21], cosmic microwave radiation [22,23], seismic activity [24,25], sunspot activity [26], atmospheric scintillation [27], temperature variability [28], meteorology [29], precipitation levels [30], streamflow and sediment movement [7,31–36], protein folding [37], finance and econophysics [38–42], electricity prices [43,44], power-grid frequency [45,46], epidemiology [47], music [48–50], ethology [51,52], multifractal harmonic signals [53], and microrheology [54].

MFDFA is a numerical algorithm designed to determine the self-similarity of a stochastic process. Putting it simply, the algorithm examines the relation between the diffusion of the process and its propagation in time or space. Auto-regressive and stochastic processes with different power-law scaling will diffuse with different rates. Fluctuation Analysis (FA), which precedes the development of MFDFA, provides a method to uncover these correlations, but fails in the presence of trends in the data, which, for example, are particularly present in weather and climate data. Detrending the data via polynomial fittings (DFA) allows one to uncover solely the relation between the inherent fluctuations and the time scaling of a process, thus circumventing the impact of non-stationarity in the data. Likewise, other methods—as empirical mode decompo-

<sup>☆</sup> The review of this paper was arranged by Prof. Hazel Andrew.

\* Corresponding author at: Forschungszentrum Jülich, Institute for Energy and Climate Research - Systems Analysis and Technology Evaluation (IEK-S TE), 52428 Jülich, Germany.

E-mail address: leonardo.rydin@gmail.com (L. Rydin Gorjão).

sition or moving average windows—are viable options to detrend the data. Another problem is that a process might be driven by more than one time scale, i.e., have more than one internal period, which can be removed either with local polynomial fittings or EMD. Moreover, a stochastic process might be of a monofractal or multifractal nature. By studying a continuum of power variations of DFA one extends into MFDFA, which permits the study of the fractality of the data by comparing power variations, i.e., a multifractal spectrum.

In this software we sought to design a computationally efficient code focused on computational speed and usability. There are currently no flexible and available implementations of MFDFA in `python`. Available are some `MATLAB` [55] as well as `R` packages [56,57]. There is a particularly thorough introductory guide to MFDFA in `MATLAB` with a source-code by Espen A. F. Ihlen [55], which is easy to implement but numerically inefficient. With this implementation efficiency was sought. This was achieved by making the most out of `python`, reshaping the code to allow for multi-threading, especially relying on `numpy`'s `polynomial`, which scales easily with modern computers having more processor cores [58]. Moreover, this library contains the most commonly applied methods alongside with DFA and MFDFA: the added feature of empirical mode decomposition is implemented to substitute the polynomial fittings; A moving window is included, especially valuable for shorter time series; The extended DFA (eDFA) method is also included, adding a second metric of fractal scaling, especially valuable for multifractal or aperiodic time series.

In the following sections we will introduce MFDFA alongside some of the aforementioned methods incorporated into the MFDFA library. We will present two classical applications, one with a monofractal process and one with a multifractal noise, and show how to use MFDFA to extract their characteristics from a single one-dimensional time series. `Python` code is presented to explicate the use of the MFDFA library. Subsequently we study two real-world time series: the sunspot time series from 1818 to 2020 which accounts for the daily recorded sunspots and the quarter-hourly electricity trading market, which accounts for a small volume of electricity sell and purchase at 15 minute windows in Continental Europe. Lastly we address a few details of the library and contribute a few closing remarks.

## 2. Theoretical basis

In the following we briefly summarise the theoretical basis of Multifractal Detrended Fluctuation Analysis. Later we detail the different included extensions and which modifications these add to the original MFDFA algorithm.

### 2.1. Multifractal detrended fluctuation analysis

Multifractal Detrended Fluctuation Analysis studies the variances of the fluctuations of a given process by considering increasing segments of a time series.

i) Take a time series  $X(t)$  (in time or space  $t$ ) with  $N$  data points, discretised as  $X_i$ ,  $i = 1, 2, \dots, N$ . Find the “detrended” profile of the process by defining

$$Y_i = \sum_{k=1}^i (X_k - \mu_X), \text{ for } i = 1, 2, \dots, N, \quad (1)$$

i.e., the cumulative sum of  $X_i$  subtracting the mean  $\mu_X$  of the data.

ii) Section the data into smaller non-overlapping segments of length  $s$ , obtaining therefore  $N_s = \text{int}(N/s)$  segments. Given the total length of the data is not always a multiple of the segment's length  $s$ , discard the last points of the data.

iii) Consider the same data, apply the same procedure, but discard now instead the first points of the data. One has now  $2N_s$  segments of the time series.

iv) To each of these segments fit a polynomial  $y_v$  of order  $m$  and calculate the variance of the difference of the data to the polynomial fit

$$F(v, s) = \frac{1}{s} \sum_{i=1}^s [Y_{(v-1)s+i} - y_{(v-1)s+i}]^2, \quad (2)$$

for  $v = 1, 2, \dots, N_s$ , where  $y_{(v-1)s+i}$  is the polynomial fitting for the segment  $Y_{(v-1)s+i}$  of length  $s$ , fitted via least-squares. The order of the polynomial  $y_v$  can be freely chosen, giving rise to the denotes (MF)DFA1, (MF)DFA2, ..., (MF)DFA $m$ , dependent on the chosen degree  $m$  of the polynomial.

v) Notice now  $F(v, s)$  is a function of each variance of each  $v$ -segment of data and of the different  $s$ -length segments chosen. Define the  $q$ -th order fluctuation function by averaging over the  $N_s$  variances of the segments of size  $s$

$$F_q(s) = \left\{ \frac{1}{N_s} \sum_{v=1}^{N_s} [F(v, s)]^{q/2} \right\}^{1/q}. \quad (3)$$

The fluctuation function  $F_q(s)$  depends on two parameters: the segment size  $s$  and the  $q$ -th power. The fluctuation function  $F_q(s)$  is the function we will focus on which the MFDFA algorithm developed extracts from the data.

Two closely related algorithms are discussed and introduced here, DFA [1] and MFDFA [3]. DFA is a particular case of MFDFA for the choice of  $q = 2$ . What is presented above is the MFDFA algorithm as according to Kantelhardt et al. [3], for which a particular choice of  $q = 2$  leads to the fluctuation function  $F_2(s)$ . The DFA fluctuation function  $F_2(s)$  can unveil solely the monofractal spectrum of a time series. If the examined time series  $X_i$  is monofractal, DFA is sufficient to describe and uncover the scaling relations in the data. If not, one must rely on MFDFA and the study of the spectrum unveiled by varying the  $q$ -th power.

We will later detail two changes: i) The first involving empirical mode decomposition (EMD) for detrending, where the local polynomial fittings are replaced and the trends of the data are subtracted by removing select Intrinsic Mode Functions (IMFs) obtained via empirical mode decomposition. ii) The second change involves substituting the non-overlapping segments with overlapping ones.

The inherent scaling properties of the data, if the data displays power-law correlations, can now be studied in a log-log plot of  $F_q(s)$  versus  $s$ , where the scaling of the data obeys a power-law with exponent  $h(q)$  as

$$F_q(s) \sim s^{h(q)} \quad (4)$$

where  $h(q)$  is the generalised Hurst exponent or *self-similarity* exponent, which will dependent on  $q$  if the data is multifractal, and relates directly to the Hurst index [59]. The generalised Hurst exponent  $h(q)$  is obtained by finding the slope of  $F_q(s)$  curve in the log-log plots.

If the data is monofractal, the generalised Hurst exponent  $h(q) = H$  is independent of  $q$  and the generalised Hurst exponent is simply the Hurst index  $H$ . On the other hand, if the data is multifractal, the dependence on  $q$  can be understood by studying the multifractal scaling exponent  $\tau(q)$ , given by

$$\tau(q) = qh(q) - 1, \quad (5)$$

which depends on the generalised Hurst exponent  $h(q)$ . Similarly, one can construct the singularity spectrum  $D(\alpha)$  as the Legendre

transform [60–63]. If  $\tau(q)$  is sufficiently smooth, the singularity strength  $\alpha$  is given by

$$\alpha = \tau'(q) = h(q) + qh'(q), \quad (6)$$

from which the singularity spectrum  $D(\alpha)$  can be constructed as

$$D(\alpha) = q\alpha - \tau(q). \quad (7)$$

The singularity spectrum  $D(\alpha)$  describes the dimension of the subset of the time series which is characterised by the singularity strength  $\alpha$  [64]. The breadth of singularity strength  $\alpha$  indicates the strength of the multifractality of the time series, centred around the most prominent scale of the time series, i.e.,  $h$ . The singularity spectrum  $D(\alpha)$  takes the shape of an inverted parabola with a maximum at  $D(\alpha = 0) = D_0$ , known as the box-counting or Minkowski–Bouligand dimension, or sometimes simply fractal dimension [60].  $D(\alpha = 1) = D_1$  is known as the information dimension and  $D(\alpha = 2) = D_2$  the correlation dimension [65]. For a clearer discussion of these properties, see Refs. [3,66]. An extensive and very illustrative representation of this can be found in Ref. [55]. For a careful analysis of the meaning and interpretation of the generalised Hurst coefficients extracted from (MF)DFA, see Ref. [67], where a description and clarification is given on what are persistent and anti-persistent motions, stationary and non-stationarity time series, among other relevant details.

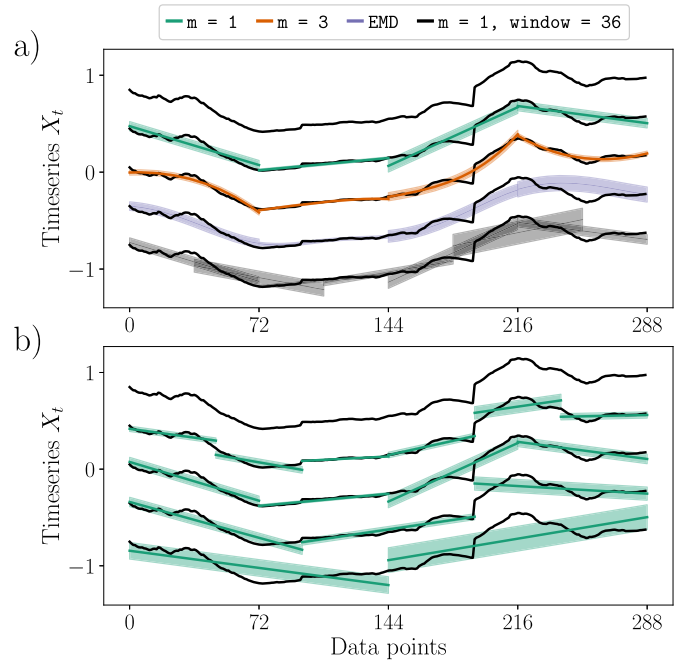
### 2.1.1. Empirical mode decomposition

Empirical mode decomposition (EMD) is a method with a variety of applications in time series analysis [68]. It seeks to extract the modes of oscillation of a time series strictly from the data. One can harness the ability of the EMD, i.e., the Hilbert–Huang decomposition of a time series, to obtain the trend or trends of the time series and utilise those to transform non-stationary into stationary data. The central concept, developed by Qian, Gu, and Zhou [6], is to substitute the detrending method employed in the traditional MFDFA, i.e., polynomial fittings, by removing instead particular Intrinsic Mode functions extracted via EMD. A sketch of the method can be seen in Fig. 1.

EMD can be summarised in a few steps: a set of intrinsic mode functions (IMFs) are extracted from the time series, obeying: 1) the number of extrema and the number of zero crossings must maximally differ by one. 2) for any point, the mean value of the envelope defined by the local extrema is zero. Numerical methods—as cubic splines—are used to find the curve that best fits “between” the local extrema of the time series. The method is applied iteratively: i) Obtain an IMF by finding the “best” curve between the local extrema of the time series; ii) Subtract this IMF to the time series; iii) Repeat. Apply the process recursively to the time series until the final IMF contains solely a residual trend of the data.

### 2.1.2. Moving windows

The overlapping moving windows included in this library is not aimed at detrending, but instead for the analyses of rather short time series or very large scales in longer time series [8]. In the literature several applications of moving average windows have been proposed as methods to remove trends and ensure stationarity, by simply removing a windowed average to the time series [8,9,42]. This is not what we do here. Here, we substitute the non-overlapping segmentation, as explain after Eq. (1), by a moving window, replacing the two separate segmentations by a moving window of each segment size  $s$  over the time series, as proposed by Zhou and Leung [8]. This is particularly relevant when examining short time series, where quickly the choice of larger lags  $s$  separates the data into a small number of segments, resulting in a poor statistics for the scaling at larger lags. A sketch of the methods can be seen in Fig. 1.



**Fig. 1.** Multifractal Detrended Fluctuation Analysis (MFDFA) of an exemplary 288 data points time series  $X_t$ . Panel a) shows, from top to bottom: the time series; a first-order polynomial fit ( $m = 1$ ); a third-order polynomial fit ( $m = 3$ ); the EMD detrending with the slowest Intrinsic Mode Function (EMD); a first-order polynomial fit with moving windows with a step of 36 data points ( $m = 1$ , window = 36). The last example with a moving window results in overlapping segments, given by the window size. For all examples, the segments have a size of  $s = 72$  data points and the lines indicate the fits, either via polynomials or EMD. Panel b) displays the changing segment size  $s$  for a first-order polynomial fit ( $m = 1$ ); From top to bottom: the time series; segmentation with  $s = 48$ ;  $s = 72$ ;  $s = 96$ ;  $s = 144$ . Shaded areas on both panels indicated the standard deviation of each segment.

### 2.1.3. Extended detrended fluctuation analysis

A new metric of similar nature as the fluctuation function  $F_q(s)$ , given in Eq. (3), has been proposed in Ref. [10]. This measure supersedes the  $q$ -order powers and takes in solely the case of DFA where  $q = 2$ . Instead of finding the average of the variances over each choice of segments of size  $s$ , it considers the difference between the extrema of the fluctuation function at each segment  $s$ . Take  $F(v, s)$  as given in Eq. (2) and extract the maximum and minimum of the variances over all windows  $v$  for a certain window size  $s$

$$\Delta F(s) = \max_v [F(v, s)] - \min_v [F(v, s)]. \quad (8)$$

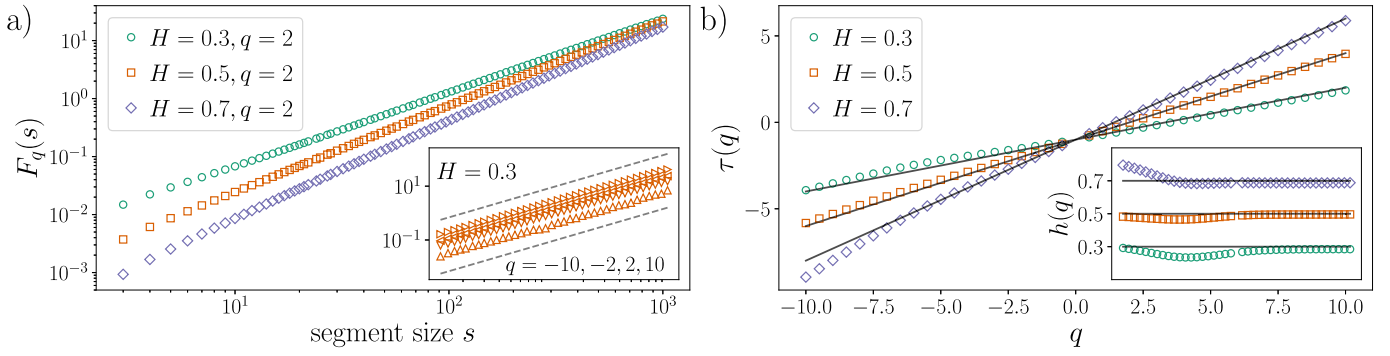
This new metric  $\Delta F(s)$  is denoted Extended Fluctuation Analysis. In general,  $\Delta F(s)$  can scale as a power law with a different exponent

$$\Delta F(s) \sim s^\beta. \quad (9)$$

This metric takes into account aperiodicities in the data which, in some sense, would be accounted for as a multifractal behaviour. It can unravel a second scaling phenomenon due to local changes of a time series' period. A purely monofractal time series would result in  $\beta = H$ .

## 3. Examples

To exemplify the usage of MFDFA, we first take two common examples of stochastic processes, a fractional Ornstein–Uhlenbeck process and general process that has a symmetric Lévy  $\alpha$ -stable



**Fig. 2.** Multifractal Detrended Fluctuation Analysis (MFDFA) of three exemplary sample paths of fractional Ornstein–Uhlenbeck processes, given by Eq. (10), with Hurst indices of  $H = 0.3, 0.5$ , and  $0.7$ . Panel a) displays the log-log plot of the segment size  $s$  versus the fluctuation function  $F_2(s)$ , given by Eq. (3), for  $q = 2$ . Each line has a slope of  $H + 1$ , as expected. The inset shows  $F_q(s)$  for the case of  $H = 0.3$  and the power variations  $q = -10, -2, 2, 10$ . These lines are all parallel indicating that the process is monofractal, as expected. The dashed lines indicate the theoretical expected scaling, i.e., a slope of  $H + 1 = 0.3 + 1$ , where the  $+1$  account for the increase in regularity due to the integration. The generalised Hurst coefficients  $h(q)$ , which are simply  $H + 1$ , are obtained by extracting the slopes of the curves (in a log-log scale). Panel b) shows the multifractal scaling exponent  $\tau(q)$ , given by Eq. (4), which exhibits a linear dependency, i.e.,  $h(q) = H$ , indicating again the process is monofractal. The processes were numerically integrated with an integration step  $\Delta t = 0.001$  over  $N = 10^4$  time units ( $N = 10^7$  data points). The MFDFA algorithm ran in  $1 \text{ min } 29 \text{ s} \pm 1.85 \text{ s}$ , for 100 segments  $s$  and 40  $q$ -variation powers, with first-order polynomial fits.

distribution, with single parameter  $\alpha$ . We will show how to extract the fluctuation function  $F_q(s)$  and how to interpret the plots conventionally extracted to perform the analysis. Subsequently we test the algorithm with real-world data on sunspot time series, following Ref. [26], and later apply the algorithm to electricity price time series from the European Power Exchange.

### 3.1. Numerically generated data

#### 3.1.1. Fractional Ornstein–Uhlenbeck process

To study the scaling effects in continuous stochastic processes, three exemplary fractional Ornstein–Uhlenbeck processes are taken, defined as [69]

$$dX_t = -\theta X_t dt + \sigma dB_t^H, \quad (10)$$

with a fractional Brownian motion  $B_t^H$  with the covariance function

$$\mathbb{E}[B_t^H B_{t'}^H] = \frac{1}{2} (|t|^{2H} + |t'|^{2H} - |t - t'|^{2H}). \quad (11)$$

Eq. (10) fixed mean reverting strength  $\theta = 1.0$  and volatility  $\sigma = 0.5$ , with three distinct Hurst indices of  $H = 0.3, 0.5$ , and  $0.7$ . Note here that the classic uncorrelated Brownian motion is given by  $H = 0.5$ . A fractional Brownian motion has a self-similarity exponent given by the Hurst index  $H$ , thus the three choices of fractional Ornstein–Uhlenbeck should result in a scaling of  $h(q) = H + 1$ . The  $+1$  is due to the integration, which smooths the fluctuations and thus increases the regularity of the process. We will now numerically integrate these processes and utilise the MFDFA library to identify the Hurst coefficients and the presence of a monofractal vs multifractal spectrum in the time series.

Let us exemplify how to numerically generate data and utilise the MFDFA library. Load the MFDFA library alongside with the fractional Brownian noise generator `fgn` included in your python console or editor.

Listing 1: Load the MFDFA library.

```
1 from MFDFA import MFDFA
2 from MFDFA import fgn
```

To numerically integrate an Ornstein–Uhlenbeck process, given by Eq. (10), we utilise an Euler–Maruyama scheme with a stepsize

$\Delta t = 0.001$  for a total time of  $t = 10^4$  (thus we have  $10^7$  data points). Here exemplified is the fractional Ornstein–Uhlenbeck process with  $H = 0.3$ .

Listing 2: Integrate Ornstein–Uhlenbeck process.

```
3 # integration time and time sampling
4 t_final = 10000
5 dt = 0.001
6 N = int(t_final/delta_t)
7
8 # The parameters theta and sigma
9 the = 1
10 sig = 0.5
11
12 # Initialise the array X
13 X = np.zeros(N)
14
15 # Generate the fractional Brownian noise
16 # with a Hurst coefficient of H = 0.3
17 dB = (t_final ** H) * fgn(N, H = 0.3)
18
19 # Integrate the process
20 for i in range(1,N):
21     X[i] = X[i-1] - the*X[i-1]*dt + sig*dB[i]
```

To retrieve the MFDFA spectrum of the generated time series, define the set of  $q$  power variations and the lags  $s$  to examine, and call the MFDFA function.

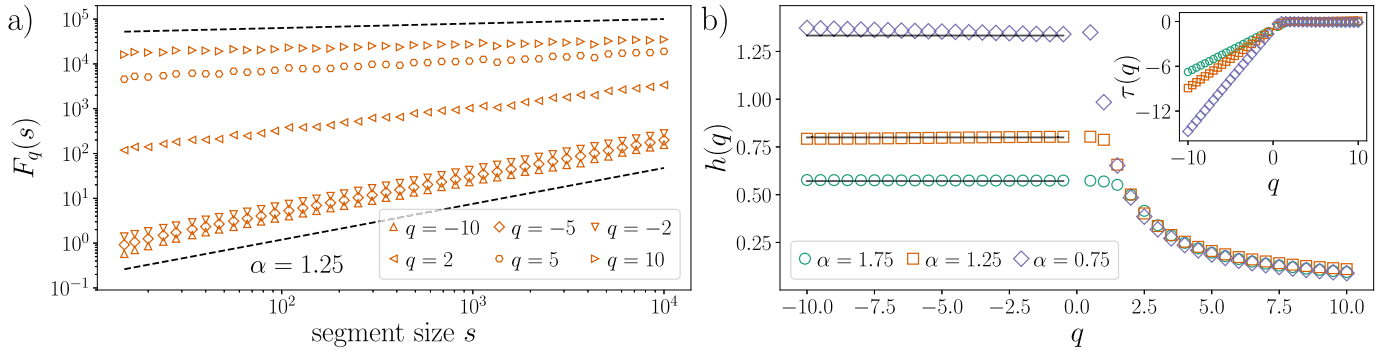
Listing 3: Applying MFDFA.

```
22 # 100 lag s points from 3 to 1000
23 lag = np.logspace(0.6, 3, 118).astype(int)
24 lag = np.unique(lag)
25
26 # q power variations, removing 0 power
27 q = np.linspace(-10, 10, 41)
28 q = q[q!=0.0]
29
30 lag, fluct = MFDFA(X, lag = lag, q = q)
```

When not declaring the values of the  $q$  powers,  $q = 2$  is assumed, thus resulting in the conventional DFA. Likewise, not declaring the order of the polynomial fitting, a first-order polynomial is assumed, i.e.,  $\text{order} = 1$ .

In Fig. 2 the MFDFA of the three processes can be seen. In panel a) the fluctuation function  $F_2(s)$ , with  $q = 2$ , is shown for a poly-





**Fig. 3.** Multifractal Detrended Fluctuation Analysis (MFDFA) of three exemplary symmetric Lévy  $\alpha$ -stable distributed processes, with  $\alpha = 1.75, 1.25$ , and  $0.75$ . In panel a) the fluctuation function  $F_q(s)$  is shown as a function of the segment size  $s$  on double logarithmic scales for  $\alpha = 1.25$  and different values of the power,  $q = -10, -5, -2, 2, 5, 10$ . For  $q > \alpha$  the curves are not parallel, indicating the multifractal nature of the process. Panel b) displays the generalised Hurst exponent  $h(q)$ , where a clear non-linear dependency on  $q$  is observable. The inset displays the multifractal scaling exponent  $\tau(q)$  displaying two clear distinct behaviours for  $q < 0$  and  $q > \alpha$ . The solid lines indicate the theoretical expected scaling for  $q < 0$ . The three processes were drawn from Lévy  $\alpha$ -stable distributions, each with  $N = 10^7$  data points. The MFDFA algorithm ran in  $1 \text{ min } 24 \text{ s} \pm 2.17 \text{ s}$  for 100 segments  $s$ , 40  $q$  powers, and third-order polynomial fits.

nomial detrending of first order. This is the conventional DFA. The slopes of each curve in the log-log plot reveal the Hurst indices of each process, i.e., the fractional Ornstein–Uhlenbeck with Hurst  $H = 0.3$  scales with a slope of  $1.3 = 0.3 + 1$ , the other two with  $H = 0.5$  and  $H = 0.7$  have a slope of  $1.5$  and  $1.7$ , respectively. The inset in panel a) shows the fluctuation function  $F_q(s)$ , with  $q = -10, -2, 2$ , and  $10$ , for the fractional Ornstein–Uhlenbeck process with  $H = 0.3$ . Note that the slope of all power variations is the same, i.e., the process is monofractal, as expected. The monofractality of the process is also evident in panel b). The multifractal scaling exponent  $\tau(q)$  is shown and is purely a linear function. Likewise, in the inset, the generalised Hurst indices  $h(q)$  for the three processes for a set of power variations  $q \in [-10, 10]$  are displayed. The linear shape of  $\tau(q)$  and constant value of  $h(q)$  in the inset indicates, as expected, that these three processes are monofractal. Small deviations are seen for very negative  $q$  powers ( $q \lesssim 7$ ), which arise due to the numeric (negative) powering operation, which highly depends on the numerical precision of the data.

### 3.1.2. Lévy-driven process

As a second example, take a collection of Lévy distributed random variables [70]. That is, each  $X_t$  is drawn independently from a symmetric  $\alpha$ -stable distribution, such that the probability density function of  $X(t)$  vanishes as a power-law  $P(x) \sim |x|^{-(\alpha+1)}$  for large  $|x|$  [70]. These processes exhibit heavy tails, ill-defined variances, and multifractal scaling. In Fig. 3 three symmetric Lévy  $\alpha$ -stable distributed processes with  $\alpha = 1.75, 1.25$ , and  $0.75$  are studied with MFDFA.

The multifractal behaviour can be identified directly in panel a), where the fluctuation function  $F_q(s)$  for  $\alpha = 1.25$  is shown. The lines of  $F_q(s)$  are not parallel for different positive  $q$  powers, showing that the process is not mono-fractal. In fact, the process is bi-fractal, having a separate behaviour for  $q < 0$  and  $q > \alpha$ . For positive power variations  $q > \alpha$ , the generalised Hurst exponent  $h(q)$  decays like  $1/q$ . For values of  $q < 0$ , the generalised Hurst exponent  $h(q) = 1/\alpha$ . This can be seen clearly in Fig. 3 b), where the generalised Hurst exponent  $h(q)$  is displayed. In the inset, one notices that the multifractal scaling exponent  $\tau(q) = 0$ , for  $q > \alpha$  (always zero for  $q > 2$ ), once again showing that none of these processes are distinguishable for positive power variations.

In general, without the aid of the multifractal spectra, which we uncovered by studying MFDFA for a range of  $q$  values, it is not possible to distinguish between Lévy distributed processes. The particular choice of  $q = 2$ , i.e., conventional DFA, obscures the fractality of these processes, as they all show a similar scaling for

$q = 2$ , i.e.,  $h(q = 2) = 1/2$  for all Lévy motions, including (non-fractional) Brownian motions (where  $\alpha = 2$ ).

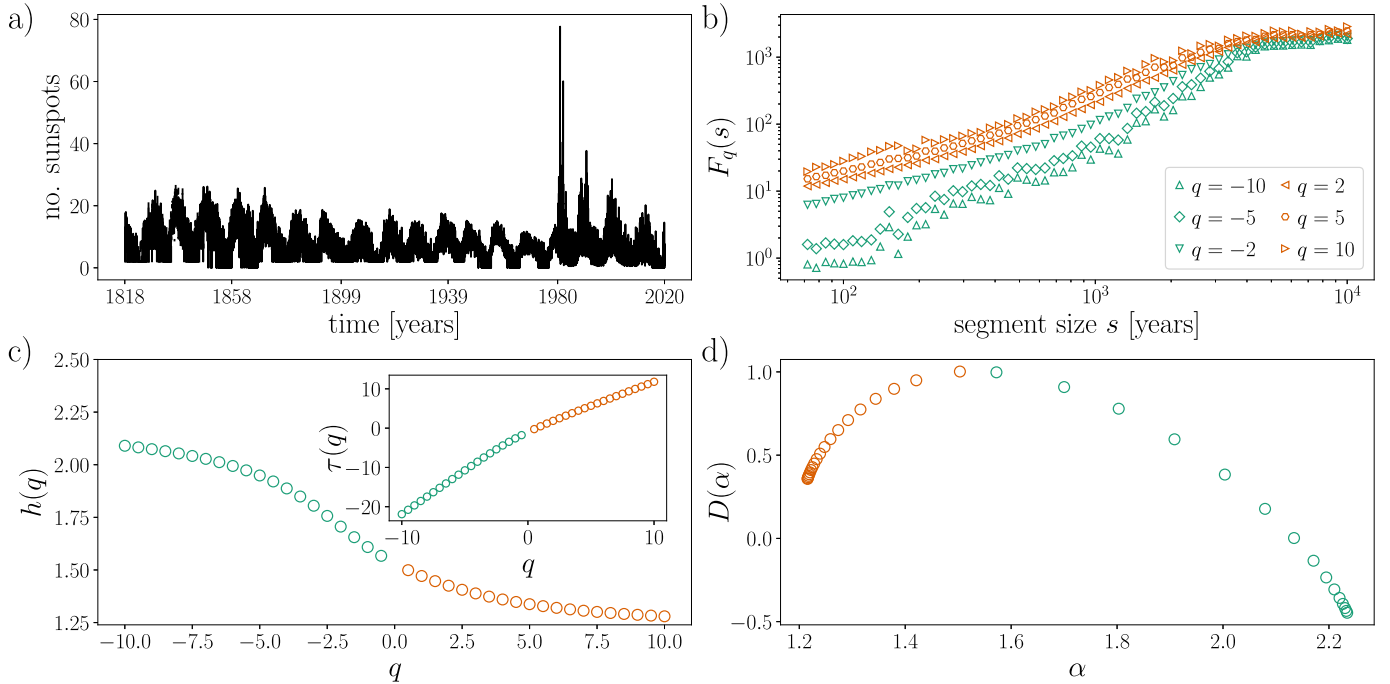
### 3.2. Real-world data, empirical mode decomposition, and extended DFA

In order to evaluate the efficiency of the algorithm, we test here two real-world data sets. Firstly, using MFDFA we will evaluate the multifractality of sunspots time series, a recurring phenomenon on the Sun's photosphere which can be observed with a telescope [26]. Secondly, we will analyse the German and Austrian spot market intraday quarter-hourly electricity price extracted from the European Power Electricity Exchange (EPEX SPOT) [71,72]. We illustrate the application of two advanced features of the developed python package, moving windows and the masking of missing data points.

#### 3.2.1. Sunspots

The sunspot numbers, also called Wolf numbers, are a rather simple measure of solar activity by counting in a weighted manner the number of groups of sunspots and single sunspots visible from the Earth in the solar photosphere, i.e. it is an integrated measure over space [74,75]. Hence, the sunspot numbers form a time series which has a mean period of about 11 years, but is far from being simply periodic [62]. Solar activity is the result of complex magneto-hydrodynamic processes in the Sun characterised by a highly complex spatio-temporal dynamics. It is of special interest to analyse the rather long series of sunspot numbers in order to explore some relations to the underlying spatio-temporal system.

The emergence of sunspots has a distinct statistics and a multifractal spectrum which has been examined in Ref. [26]. This publication has become a reference for multifractal studies as the data from the ILSO World Data Center, Royal Observatory of Belgium, Brussels is freely available [73]. Here, we will focus on numerical efficiency and how to deal with missing or corrupt data. We utilised another feature integrated in MFDFA that enables an efficient management of missing data points. In python's numpy arrays, missing or corrupt values in a time series can be handled with masked data, which logs the missing data points and takes these into account while performing averages, sums, and power operations. When calculating averages or the variance of a segment, or when taking powers, the masked entries are not taken into account. For the particular application with sunspot time series, which are recorded daily since 1818, there are 3247 missing values, over a total of 74145 entries, i.e., roughly 4.4% of the data is missing. To go around this, simply use



**Fig. 4.** Multifractal Detrended Fluctuation Analysis (MF DFA) of sunspot time series from 1818 to 2020, by the ILSO World Data Center, Royal Observatory of Belgium, Brussels [73]. Panel a) shows the number of sunspots registered from 1818 to 2020. Panel b) displays the fluctuation function  $F_q(s)$  as a function of the segment size  $s$  on a double-logarithmic scale for  $q = -10, -5, -2, 2, 5, 10$ , with positive  $q$  values in orange and negative  $q$  values in green. Panel c) displays the generalised Hurst coefficient  $h(q)$  over  $q$ , and the inset displays the multifractal scaling exponent  $\tau(q)$ , given by Eq. (4), both highlighting the multifractal spectrum of the data ( $h(q)$  is not constant over  $q$ ,  $\tau(q)$  is not linear over  $q$ ). Panel d) displays the singularity spectrum  $D(\alpha)$  over the singularity strength  $\alpha$  which shows a large breadth of  $\alpha$  spanning over  $[1.25, 2.25]$ , indicating the strong multifractality of the data. The MF DFA algorithm ran in  $426 \text{ ms} \pm 11.8 \text{ ms}$ , for 70 segments  $s$ , 40  $q$  powers, and first-order polynomial fits. The missing values were neatly removed by utilising numpy's masked arrays, which is integrated in MF DFA and allows the user to simply "mask" empty or corrupted data. (For interpretation of the colours in the figure(s), the reader is referred to the web version of this article.)

#### Listing 4: MF DFA and missing data.

```

1 # Read data whichever way preferred
2 data = read_data('sunspot.csv')
3
4 # Mask missing values. For this case -1
5 # is a missing entry in the record
6 data[data == -1.] = np.nan
7 data = np.ma.masked_invalid(data)
8
9 # Run MF DFA (choose lag and q)
10 lag, fluct = MF DFA(data, lag, q)

```

The MF DFA will extract the variances as it is possible, taking into account the missing values in the time series. Here we highlight that MF DFA calculated 40  $q$ -powers over 70 segments  $s$  in  $426 \text{ ms} \pm 11.8 \text{ ms}$ .

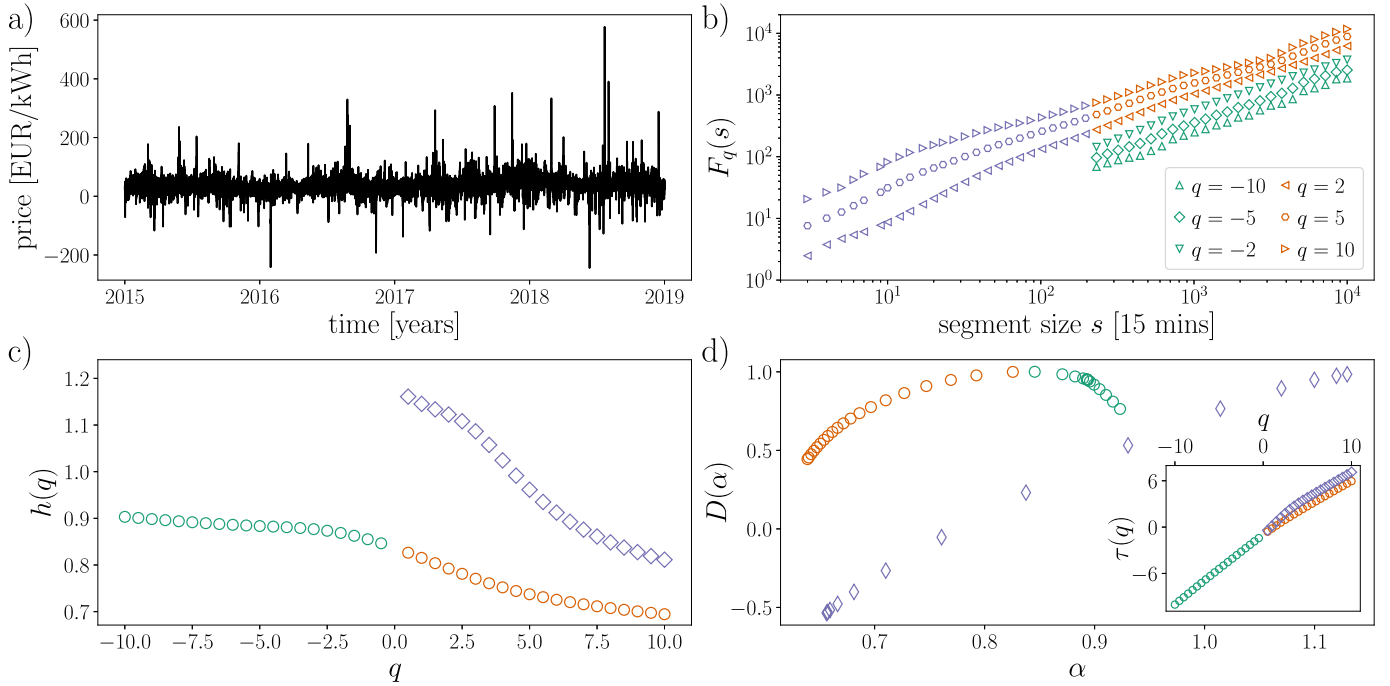
In Fig. 4 we display the fluctuation function  $F_q(s)$  for  $q = -10, -5, -2, 2, 5, 10$ , in panel b), for  $s \in [70, 1000]$ . These curves are not parallel, suggesting the time series is not monofractal. In panel c) the generalised Hurst exponent  $h(q)$  is shown as function of  $q$ , and similarly the multifractal scaling exponent  $\tau(q)$  in inset. The generalised Hurst exponent  $h(q)$  is not constant over  $q$  and consequently the multifractal scaling exponent  $\tau(q)$  is not linear, indicating clearly the time series in multifractal. In panel d) we display the singularity spectrum  $D(\alpha)$  over the singularity strength  $\alpha$ , as given by Eq. (7). The singularity strength  $\alpha$  spans a wide range of values, over  $[1.25, 2.25]$ , indicating how strongly multifractal the time series is. Here recall that a monofractal time series, as the fractional Ornstein–Uhlenbeck previously shown in Fig. 2, has a very narrow range of the singularity strength  $\alpha$ , centred around  $H$ . For the case of sunspot time series we see a wide range of  $\alpha$ , indicating the various scales of the phenomenon. Note

as well that  $h(q)$  and  $\alpha$  are always larger than 1, indicating that this is a non-stationary process.

#### 3.2.2. German and Austrian spot market intraday quarter-hourly electricity price time series

We will examine now a 4-year long time series sampled at 15 minutes of the spot market intraday quarter-hourly German and Austrian electricity price [71,72], from the 1st of January 2015 to end of December 2019, traded at the European Power Exchange (EPEX SPOT). To the extent of our knowledge no multifractal analysis of this particular data has been performed before, but other multifractal studies of price time series exist [44]. In Wang et al. [44], the authors examine different scaling properties for selected periods of low, regular, and high electricity price for some United States of America's electricity markets in the year 2000 and 2001. Here we propose a different analysis, studying the data and examining a short and long time scale of the data without separating different activity periods.

We know that the 15 minute trading electricity market amounts to a small volume of the overall exchange electricity sold, thus this market serves only electricity producers which can either extract or inject power from the power-grid system in a very fast manner ( $< 15$  minutes) [76,77]. This will lead us to explore to separate scaling phenomena in the data: A short and a long timescale of market activities. The expectation is that the very short-time trading is highly volatile, given the necessity of the power grid in injecting or extracting power is a fairly speedy manner. In the long run, the quarter-hourly market is intrinsically linked to the larger hourly and daily electricity market, which has far less variability, as most of the power is sold in lengthier contracts, stabilising the value of the electricity price. Thus one expects a narrower multifractality at large temporal scales. Multifractality is nevertheless expected, as the system exhibits very large yet seldom negative



**Fig. 5.** Multifractal Detrended Fluctuation Analysis (MFDFA) of the spot market intraday quarter-hourly German and Austrian electricity price time series from 2015 to 2019, traded at the European Power Exchange (EPEX SPOT) [71,72]. Panel a) displays the price in EUR/kWh from 2015 to 2019. Panel b) displays the log-log plot of the segment size  $s$  versus the fluctuation function  $F_q(s)$  for  $q = -10, -5, -2, 2, 5, 10$ . Orange and green markers indicate the segments larger than two days, where purple indicate segments between 1 and 48 hours. There two scales are studied separately. Panel c) displays the generalised Hurst coefficient  $h(q)$  over  $q$ , and the inset displays the multifractal scaling exponent  $\tau(q)$ , given by Eq. (4). Panel d) displays the singularity spectrum  $D(\alpha)$  over the singularity strength  $\alpha$ , indicating precisely the high volatility of the market at short time scales. In comparison, the longer time scales ( $> 48$  hours) are much “milder”, and the variations of  $\alpha \in [0.67, 0.92]$ , which indicates the process is both stationary, in the long run, and only moderately volatile. The MFDFA algorithm ran in  $2 \text{ min } 11 \text{ s} \pm 4.43 \text{ s}$  for 50 segments  $s$ , 40  $q$  powers, first-order polynomial fits, and the moving window.

prices, as well as an occasional four of five-fold increase of the (positive) prices, again occurring seldom and lasting very short periods.

In order to obtain a better statistics of the shorter time scales, we will employ MFDFA’s moving windows method previously discussed. The moving window method requires the input of the number of steps used to “move” the windows. For the following example, the window parameter is set to 1, thus each overlapping window is displaced by solely 1 data point. This substantially increases the computational time as each averaging operation is repeated by the number of segments  $s - 1$ . For Fig. 5, the total calculation lasted  $2 \text{ min } 11 \text{ s} \pm 4.43 \text{ s}$  for the windowed mode, compared with  $764 \text{ ms} \pm 9.32 \text{ ms}$  with the conventional non-overlapping windows, for 50  $s$  segments.

Listing 5: MFDFA and moving window.

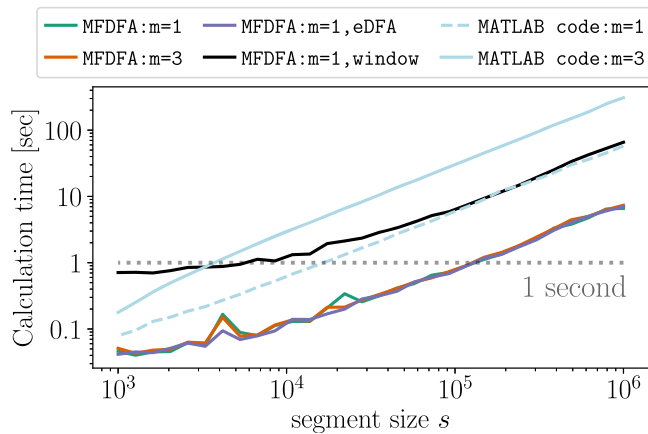
```
1 # Read data whichever way preferred
2 data = read_data('price.csv')
3
4 # Run MFDFA (choose lag and q)
5 lag, fluct = MFDFA(data, lag, q,
6                   extension = {'window' : 1})
```

In Fig. 5 we display the MFDFA analysis of the price time series, as previously done for the sunspots in Fig. 4. We perform a similar analysis as above, thus we will condense the technical details and focus on the interpretation. Previous studies point to a clear separation of the scaling behaviour of price time series [43,78,79]. They separate two time scales for periods shorter and longer than 24 hours. These studies used pricing data from before 2004 for the Nordic grid (Nordpool). In our analysis, we similarly separate two scales, between 1 and 48 hours and between 48 hours and

10 days. These are indicated in purple (short timescale) and orange and green (long time scale). We first observe that negative  $q$  powers do not exist for the short time scale. This is not unusual, many processes do not show a multifractal spectrum with negative  $q$  values. Note that this involves taking negative powers of the average of the variances, which is not always well defined for short  $s$  segments. This also served as a threshold to assess the change in the fractal behaviour of the time series. For the large time scales ( $> 48$  hours) the negative powers are well defined, and we can identify the full singularity spectrum  $D(\alpha)$ , as seen in panel d). We note that the short time scale ( $< 48$  hours) has a very strong multifractality (in purple). The singularity strength  $\alpha$ , which we can only extract for positive  $q$  values, has a very large breadth, especially with its equivalent for the large time scale (in orange). This is well grounded on the previous arguments of having a very volatile market at these short time scales, thus these results are in line with what is known about this market: The high volatility and occasional burst—into very large electricity prices or into negative prices—generate a wide range of the singularity strength. The long-term stability, connected with the larger intraday and day-ahead electricity markets, makes the process far less multifractal at large temporal scales.

#### 4. The MFDFA library

The Multifractal Detrended Fluctuation Analysis library MFDFA in python presented is a standalone package based integrally on python’s numpy [58]. It can be found in <https://github.com/LRydin/MFDFA>. It harnesses numpy’s vectorised polynomial fittings, making it possible to utilise all computational cores in a computer’s processor(s). Additionally, EMD is included as an extra feature, which is integrated into MFDFA by simply installing the python library PyEMD [80]. The conventional plots associated



**Fig. 6.** Speed performance of MFDDFA for time series with sizes varying between  $[10^3, 10^6]$  data points of a fractional Ornstein–Uhlenbeck as given by Eq. (10). Included are first-order and third-order polynomial fits, first-order fits with extended DFA, and first-order fits with a moving window with a step size of 5. A comparison with the distributed MATLAB code is included [55]. Tests ran on python 3.8.2 and MATLAB R2020b. MFDDFA has a average speed-up compared with the MATLAB code, with a five-fold speed increase for first-order polynomial fits ( $m=1$ ) and a  $\times 27$ -fold increase for third-order polynomials fits ( $m=3$ ). Both codes were tested for 100 segments  $s$  and 40  $q$  powers. All tasks were performed on a laptop on two computer cores at 2.9GHz each.

with multifractal analysis, i.e.,  $F_q(s)$  vs.  $s$ ,  $h(q)$  and  $\tau(q)$  vs.  $q$ , and  $D(\alpha)$  vs.  $\alpha$ , are available as well and require the plotting library matplotlib [81].

The MFDDFA library accepts numpy's masked arrays, which is particularly convenient when dealing with time series with missing data, as exemplified in Sec. 3.2 and Fig. 4.

The MFDDFA library offers a considerable speed-up in comparison with the available MATLAB version. The library is fully developed to work with multi-threading, which shows an increase in the performance, while handling time series larger than  $10^5$  data points. In Fig. 6 we display the performance of the MFDDFA library for time series of fractional Ornstein–Uhlenbeck processes given in Eq. (10) of increasing length. The MFDDFA operation scales linearly with the number of points of the generated time series. The MFDDFA algorithm runs in under 1 s for time series having up to  $10^5$  datapoints, with a first-order polynomial fittings, 100 segments  $s$  and 40  $q$  powers, and outperforms the conventional library in MATLAB by up to a factor of  $\times 10^3$  in computational speed.

Estimation error and significance calculation have not been included in the library, as the focus lied on computational speed and the inclusion of several extra features, as discussed.

## 5. Conclusion

We have presented a numerically efficient python implementation of Multifractal Detrended Fluctuation Analysis called MFDDFA. MFDDFA has found extensive application in the past two decades, yet a reliable, all-encompassing open-source software in python does not exist to this date. In this library we have harnessed the most of python's flexibility with handling matricial operations and multi-threaded polynomial fittings. In this implementation we have included some of the more common extensions of MFDDFA, including a simple empirical mode decomposition as a mechanism to detrend the data, a moving window to handle very short time series, and the extended Detrended Fluctuation Analysis, which can track a different scaling mechanism for non-stationary time series. The MFDDFA library can also handle missing values in the data with the aid of numpy's masked time series.

We have initially turned to two classic numerically generated stochastic processes, fractional Ornstein–Uhlenbeck processes and

Lévy-distributed motions, and uncovered their monofractal and multifractal with MFDDFA. Subsequently we have studied two real-world time series, the sunspot count from 1818 to 2020 and the quarter-hourly electricity price time series from 2015 to 2019. For both we performed a multifractal analysis, unveiling their scaling properties and the strength of their multifractality. We focused on MFDDFA's speed, the ability to handled missing data, and the integrated overlapping moving window. The analysis displayed here covered only part of MFDDFA's integrated options, thus we leave the user to explore the other implemented methods, as the extended DFA and EMD detrending, as these are more specialised to particular research fields.

We hope with this contribution we open a door to fast MFDDFA calculations that can be performed on a local machine without an extensive numerical effort and very long time runs, thus permitting in the future to analyse larger time series.

## Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

## Acknowledgements

L.R.G. kindly thanks Francisco Meirinhos for all the help with python, and Fabian Harang, Marc Lagunas Merino, Anton Yurchenko-Tytarenko, Dennis Schroeder, Michele Giordano, Giulia di Nunno, and Fred Espen Benth for their support. L.R.G. and D.W. gratefully acknowledge support by the Helmholtz Association, via the joint initiative *Energy System 2050 - A Contribution of the Research Field Energy* and the grant *Uncertainty Quantification - From Data to Reliable Knowledge (UQ)*, with grant no. ZT-I-0029, the scholarship funding from *E.ON Stipendienfonds*, and the *STORM - Stochastics for Time-Space Risk Models* project of the Research Council of Norway (RCN) no. 274410. This work was performed as part of the Helmholtz School for Data Science in Life, Earth and Energy (HDS-LEE). J.K. was financed by the Ministry of Science and Higher Education of the Russian Federation within the framework of state support for the creation and development of World-Class Research Center "Digital biodesign and personalised healthcare", no. 075-15-2020-926.

## References

- [1] C.-K. Peng, S.V. Buldyrev, S. Havlin, M. Simons, H.E. Stanley, A.L. Goldberger, Phys. Rev. E 49 (1994) 1685–1689, <https://doi.org/10.1103/PhysRevE.49.1685>.
- [2] C. Peng, S. Havlin, H.E. Stanley, A.L. Goldberger, Chaos 5 (1) (1995) 82–87, <https://doi.org/10.1063/1.166141>.
- [3] J.W. Kantelhardt, S.A. Zschiegner, E. Koscielny-Bunde, S. Havlin, A. Bunde, H. Stanley, Physica A 316 (1) (2002) 87–114, [https://doi.org/10.1016/S0378-4371\(02\)01383-3](https://doi.org/10.1016/S0378-4371(02)01383-3).
- [4] Z. Wu, N.E. Huang, Proc. R. Soc. A, Math. Phys. Eng. Sci. 460 (2046) (2004) 1597–1611, <https://doi.org/10.1098/rspa.2003.1221>.
- [5] Z. Wu, N.E. Huang, Adv. Adapt. Data Anal. 01 (01) (2009) 1–41, <https://doi.org/10.1142/S1793536909000047>.
- [6] X.-Y. Qian, G.-F. Gu, W.-X. Zhou, Physica A 390 (23) (2011) 4388–4395, <https://doi.org/10.1016/j.physa.2011.07.008>.
- [7] X. Zhang, G. Zhang, L. Qiu, B. Zhang, Y. Sun, Z. Gui, Q. Zhang, Water 11 (5) (2019) 891, <https://doi.org/10.3390/w11050891>.
- [8] Y. Zhou, Y. Leung, J. Stat. Mech. Theory Exp. 2010 (06) (2010) P06021, <https://doi.org/10.1088/1742-5468/2010/06/p06021>.
- [9] S. Lai, L. Wan, X. Zeng, J. Phys. Conf. Ser. 1345 (2019) 042086, <https://doi.org/10.1088/1742-6596/1345/4/042086>.
- [10] A.N. Pavlov, A.S. Abdurashitov, A.A. Koronovskii, O.N. Pavlova, O.V. Semyachkina-Glushkovskaya, J. Kurths, Commun. Nonlinear Sci. Numer. Simul. 85 (2020) 105232, <https://doi.org/10.1016/j.cnsns.2020.105232>.
- [11] A.N. Pavlov, A.I. Dubrovsky, A.A. Koronovskii, O.N. Pavlova, O.V. Semyachkina-Glushkovskaya, J. Kurths, Chaos 30 (7) (2020) 073138, <https://doi.org/10.1063/5.0011823>.



- [12] A. Pavlov, A. Dubrovsky, A. Koronovskii Jr., O. Pavlova, O. Semyachkina-Glushkovskaya, J. Kurths, *Chaos Solitons Fractals* 139 (2020) 109989, <https://doi.org/10.1016/j.chaos.2020.109989>.
- [13] A.N. Pavlov, O.N. Pavlova, O.V. Semyachkina-Glushkovskaya, J. Kurths, *Eur. Phys. J. Plus* 136 (1) (2021) 10, <https://doi.org/10.1140/epjp/s13360-020-00980-x>.
- [14] B. Podobnik, H.E. Stanley, *Phys. Rev. Lett.* 100 (2008) 084102, <https://doi.org/10.1103/PhysRevLett.100.084102>.
- [15] W.-X. Zhou, *Phys. Rev. E* 77 (2008) 066211, <https://doi.org/10.1103/PhysRevE.77.066211>.
- [16] J. Alvarez-Ramirez, E. Rodriguez, J.C. Echeverria, *Phys. Rev. E* 79 (2009) 057202, <https://doi.org/10.1103/PhysRevE.79.057202>.
- [17] C. Chianca, A. Ticona, T. Penna, *Physica A* 357 (3) (2005) 447–454, <https://doi.org/10.1016/j.physa.2005.03.047>.
- [18] K. Hu, P.C. Ivanov, Z. Chen, P. Carpena, H. Eugene Stanley, *Phys. Rev. E* 64 (2001) 011114, <https://doi.org/10.1103/PhysRevE.64.011114>.
- [19] P.C. Ivanov, L.A.N. Amaral, A.L. Goldberger, S. Havlin, M.G. Rosenblum, Z.R. Struzik, H.E. Stanley, *Nature* 399 (6735) (1999) 461, <https://doi.org/10.1038/20924>.
- [20] S. Dutta, D. Ghosh, S. Chatterjee, *Front. Physiol.* 4 (2013) 274, <https://doi.org/10.3389/fphys.2013.00274>.
- [21] A. Madanchi, F. Taghavi-Shahri, S.M. Taghavi-Shahri, M.R. Rahimi Tabar, *Eur. Phys. J. B* 93 (7) (2020) 126, <https://doi.org/10.1140/epjb/e2020-100561-4>.
- [22] M.S. Movahed, F. Ghasemi, S. Rahvar, M.R.R. Tabar, *Phys. Rev. E* 84 (2011) 021103, <https://doi.org/10.1103/PhysRevE.84.021103>.
- [23] M.S. Movahed, B. Javanmardi, R.K. Sheth, *Mon. Not. R. Astron. Soc.* 434 (4) (2013) 3597–3605, <https://doi.org/10.1093/mnras/stt1284>.
- [24] L. Telesca, V. Lapenna, M. Macchiato, *Physica A* 354 (2005) 629–640, <https://doi.org/10.1016/j.physa.2005.02.053>.
- [25] S. Shadkhoo, F. Ghanbarnejad, G. Jafari, M.R.R. Tabar, *Open Phys.* 7 (3) (2009) 620–623, <https://doi.org/10.2478/s11534-009-0058-0>.
- [26] M. Sadeq Movahed, G.R. Jafari, F. Ghasemi, S. Rahvar, M.R.R. Tabar, *J. Stat. Mech. Theory Exp.* 2006 (2) (2006) P02003, <https://doi.org/10.1088/1742-5468/2006/02/p02003>.
- [27] H.J. Tanna, K.N. Pathak, *Astrophys. Space Sci.* 350 (1) (2014) 47–56, <https://doi.org/10.1007/s10509-013-1742-5>.
- [28] P.G. Meyer, H. Kantz, *Clim. Dyn.* 53 (11) (2019) 6909–6917, <https://doi.org/10.1007/s00382-019-04965-0>.
- [29] I.T. Pedron, *J. Phys. Conf. Ser.* 246 (2010) 012034, <https://doi.org/10.1088/1742-6596/246/1/012034>.
- [30] Y. Tessier, S. Lovejoy, P. Hubert, D. Schertzer, S. Pecknold, J. Geophys. Res., *Atmos.* 101 (D21) (1996) 26427–26440, <https://doi.org/10.1029/96JD01799>.
- [31] C. Matsoukas, S. Islam, I. Rodriguez-Iturbe, *J. Geophys. Res., Atmos.* 105 (D23) (2000) 29165–29172, <https://doi.org/10.1029/2000JD900419>.
- [32] D. Koutsoyiannis, *Hydrol. Sci. J.* 48 (1) (2003) 3–24, <https://doi.org/10.1623/hysj.48.1.3.43481>.
- [33] J.W. Kantelhardt, E. Koscielny-Bunde, D. Rybski, P. Braun, A. Bunde, S. Havlin, *J. Geophys. Res., Atmos.* 111 (D1) (2006) D01106, <https://doi.org/10.1029/2005JD005881>.
- [34] Q. Zhang, C.-Y. Xu, Y.D. Chen, Z. Yu, *Hydrol. Process.* 22 (26) (2008) 4997–5003, <https://doi.org/10.1002/hyp.7119>.
- [35] R. Rodríguez, M.C. Casas, A. Redaño, *Meteorol. Atmos. Phys.* 121 (3) (2013) 181–187, <https://doi.org/10.1007/s00703-013-0256-6>.
- [36] Y. Wu, Y. He, M. Wu, C. Lu, S. Gao, Y. Xu, *Sci. Rep.* 8 (1) (2018) 16553, <https://doi.org/10.1038/s41598-018-35032-z>.
- [37] P. Figueirêdo, E. Nogueira, M. Moret, S. Coutinho, *Physica A* 389 (10) (2010) 2090–2095, <https://doi.org/10.1016/j.physa.2009.11.045>.
- [38] L. Zunino, B. Tabak, A. Figliola, D. Pérez, M. Garavaglia, O. Rosso, *Physica A* 387 (26) (2008) 6558–6566, <https://doi.org/10.1016/j.physa.2008.08.028>.
- [39] L. Zunino, A. Figliola, B.M. Tabak, D.G. Pérez, M. Garavaglia, O.A. Rosso, *Chaos Solitons Fractals* 41 (5) (2009) 2331–2340, <https://doi.org/10.1016/j.chaos.2008.09.013>.
- [40] D. Grech, G. Pamula, *Acta Phys. Pol. A* 123 (2013) 529–537, <https://doi.org/10.12693/aphyspola.123.529>.
- [41] S. Drożdż, R. Kowalski, P. Oświecimka, R. Rak, R. Gębarowski, *Complexity* 2018 (2018) 7015721, <https://doi.org/10.1155/2018/7015721>.
- [42] M. Lee, J.W. Song, S. Kim, W. Chang, *Physica A* 512 (2018) 1278–1294, <https://doi.org/10.1016/j.physa.2018.08.030>.
- [43] R. Weron, I. Simonsen, P. Wilman, in: H. Takayasu (Ed.), *The Application of Econophysics*, Springer Japan, Tokyo, 2004, pp. 182–191.
- [44] F. Wang, G. Liao, J. Li, X. Li, T. Zhou, *Physica A* 392 (22) (2013) 5723–5734, <https://doi.org/10.1016/j.physa.2013.07.039>.
- [45] L. Shalalfah, P. Bogdan, E. Jonckheere, in: 2016 IEEE International Conference on Smart Grid Communications (SmartGridComm), 2016, pp. 466–471.
- [46] L. Rydin Gorjão, R. Jumar, H. Maass, V. Hagenmeyer, G.C. Yalcin, J. Kruse, M. Timme, C. Beck, D. Witthaut, B. Schäfer, *Nat. Commun.* 11 (1) (2020) 6362, <https://doi.org/10.1038/s41467-020-19732-7>.
- [47] Y. Leung, E. Ge, Z. Yu, *Ann. Assoc. Am. Geogr.* 101 (6) (2011) 1221–1240, <https://doi.org/10.1080/00045608.2011.592733>.
- [48] G.R. Jafari, P. Pedram, L. Hedayatfar, *J. Stat. Mech. Theory Exp.* 2007 (04) (2007) P04012, <https://doi.org/10.1088/1742-5468/2007/04/p04012>.
- [49] L. Telesca, M. Lovallo, *Proc. R. Soc. A, Math. Phys. Eng. Sci.* 467 (2134) (2011) 3022–3032, <https://doi.org/10.1098/rspa.2011.0118>.
- [50] H.V. Ribeiro, L. Zunino, R.S. Mendes, E.K. Lenzi, *Physica A* 391 (7) (2012) 2421–2428, <https://doi.org/10.1016/j.physa.2011.12.009>.
- [51] C.L. Alados, M.A. Huffman, *Ethology* 106 (2) (2000) 105–116, <https://doi.org/10.1046/j.1439-0310.2000.00497.x>.
- [52] K.M. Rutherford, M.J. Haskell, C. Glasbey, R. Jones, A.B. Lawrence, *Appl. Anim. Behav. Sci.* 83 (2) (2003) 125–139, [https://doi.org/10.1016/S0168-1591\(03\)00115-1](https://doi.org/10.1016/S0168-1591(03)00115-1).
- [53] J. Li, X. Ma, M. Zhao, X. Cheng, *Electronics* 8 (2) (2019) 209, <https://doi.org/10.3390/electronics8020209>.
- [54] A. Madanchi, J.W. Yu, M.R.R. Tabar, W.B. Lee, S.E.E. Rahbari, *Soft Matter* (2021), <https://doi.org/10.1039/D0SM02039G>. Accepted Publication.
- [55] E. Ihlen, *Front. Physiol.* 3 (2012) 141, <https://doi.org/10.3389/fphys.2012.00141>.
- [56] M. Laib, L. Telesca, M. Kanevski, *Chaos* 28 (3) (2018) 033108, <https://doi.org/10.1063/1.5022737>.
- [57] M. Laib, J. Golay, L. Telesca, M. Kanevski, *Chaos Solitons Fractals* 109 (2018) 118–127, <https://doi.org/10.1016/j.chaos.2018.02.024>.
- [58] C.R. Harris, K.J. Millman, S.J. van der Walt, R. Gommers, P. Virtanen, D. Cournapeau, E. Wieser, J. Taylor, S. Berg, N.J. Smith, R. Kern, M. Picus, S. Hoyer, M.H. van Kerkwijk, M. Brett, A. Haldane, J.F. del Río, M. Wiebe, P. Peterson, P. G. Erard-Marchant, K. Sheppard, T. Reddy, W. Weckesser, H. Abbasi, C. Gohlke, T.E. Oliphant, *Nature* 585 (7825) (2020) 357–362, <https://doi.org/10.1038/s41586-020-2649-2>.
- [59] H.E. Hurst, *Trans. Am. Soc. Civ. Eng.* 116 (1951) 770–799, <https://doi.org/10.1061/TACEAT.0006518>.
- [60] H. Hentschel, I. Procaccia, *Physica D* 8 (3) (1983) 435–444, [https://doi.org/10.1016/0167-2789\(83\)90235-X](https://doi.org/10.1016/0167-2789(83)90235-X).
- [61] T.C. Halsey, M.H. Jensen, L.P. Kadanoff, I. Procaccia, B.I. Shraiman, *Phys. Rev. A* 33 (1986) 1141–1151, <https://doi.org/10.1103/PhysRevA.33.1141>.
- [62] J. Kurths, H. Herzel, *Physica D* 25 (1) (1987) 165–172, [https://doi.org/10.1016/0167-2789\(87\)90099-6](https://doi.org/10.1016/0167-2789(87)90099-6).
- [63] C. Meneveau, K. Sreenivasan, *Nucl. Phys. B, Proc. Suppl.* 2 (1987) 49–76, [https://doi.org/10.1016/0920-5632\(87\)90008-9](https://doi.org/10.1016/0920-5632(87)90008-9).
- [64] H. Salat, R. Murcio, E. Arcaute, *Physica A* 473 (2017) 467–487, <https://doi.org/10.1016/j.physa.2017.01.041>.
- [65] K. Falconer, *Fractal Geometry: Mathematical Foundations and Applications*, 3rd edition, John Wiley & Sons, Chichester, 2014.
- [66] A.-L. Barabási, T. Vicsek, *Phys. Rev. A* 44 (1991) 2730–2733, <https://doi.org/10.1103/PhysRevA.44.2730>.
- [67] F. Serinaldi, *Physica A* 389 (14) (2010) 2770–2781, <https://doi.org/10.1016/j.physa.2010.02.044>.
- [68] N.E. Huang, Z. Shen, S.R. Long, M.C. Wu, H.H. Shih, Q. Zheng, N.-C. Yen, C.C. Tung, H.H. Liu, *Proc. R. Soc. A, Math. Phys. Eng. Sci.* 454 (1971) (1998) 903–995, <https://doi.org/10.1098/rspa.1998.0193>.
- [69] M.R.R. Tabar, *Analysis and Data-Based Reconstruction of Complex Nonlinear Dynamical Systems*, Springer International Publishing, 2019.
- [70] D. Applebaum, *Lévy Processes and Stochastic Calculus*, 2nd edition, Cambridge University Press, 2011.
- [71] European Power Exchange (EPEX SPOT), Market data, <https://www.epexspot.com/en/market-data>, 2021.
- [72] Fraunhofer Institute for Solar Energy System ISE, Energy-Charts, Energy-Charts. 2015–2019, [https://energy-charts.info/charts/price\\_spot\\_market/chart.htm](https://energy-charts.info/charts/price_spot_market/chart.htm), 2020.
- [73] SILSO World Data Center, Royal Observatory of Belgium, Brussels, the International Sunspot Number, international Sunspot Number Monthly Bulletin and online catalogue: 1818–2020 <http://www.sidc.be/silso/>, 2020.
- [74] M. Stix, *The Sun*, 2nd edition, Springer-Verlag Berlin Heidelberg, 2002.
- [75] A. Balogh, H. Hudson, K. Petrovay, R. von Steiger (Eds.), *The Solar Activity Cycle*, 1st edition, Springer-Verlag, New York, 2015.
- [76] S.M. Braun, C. Brunner, *Z. Energiewirtschaft.* 42 (3) (2018) 257–270, <https://doi.org/10.1007/s12398-018-0228-0>.
- [77] M. Narajewski, F. Ziel, *J. Commodity Mark.* 19 (2020) 100107, <https://doi.org/10.1016/j.jcomm.2019.100107>.
- [78] I. Simonsen, *Physica A* 322 (2003) 597–606, [https://doi.org/10.1016/S0378-4371\(02\)01938-6](https://doi.org/10.1016/S0378-4371(02)01938-6).
- [79] R. Weron, M. Bierbrauer, S. Trück, in: *Proceedings of the XVIII Max Born Symposium “Statistical Physics Outside Physics”*, *Physica A* 336 (1) (2004) 39–48.
- [80] D. Laszuk, Python implementation of empirical mode decomposition algorithm, <https://github.com/laszukdawid/PyEMD>, 2017.
- [81] J.D. Hunter, *Comput. Sci. Eng.* 9 (3) (2007) 90–95, <https://doi.org/10.1109/MCSE.2007.55>.