

Multifractional Brownian motion characterization based on Hurst exponent estimation and statistical learning

Cite as: Chaos 32, 083148 (2022); doi: 10.1063/5.0093836

Submitted: 30 March 2022 · Accepted: 2 August 2022 ·

Published Online: 29 August 2022



View Online



Export Citation



CrossMark

Dawid Szarek,^{1,a)} Ireneusz Jabłoński,² Diego Krapf,³ and Agnieszka Wyłomańska¹

AFFILIATIONS

¹Chair of Applied Mathematics, Faculty of Pure and Applied Mathematics, Hugo Steinhaus Center, Wrocław University of Science and Technology, Wyspiańskiego 27, 50-370 Wrocław, Poland

²Chair of Electronic and Photonic Metrology, Faculty of Electronics, Photonics and Microsystems, Wrocław University of Science and Technology, B. Prusa 53/55, 50-317 Wrocław, Poland

³Department of Electrical and Computer Engineering, Colorado State University, Fort Collins, Colorado 80523, USA

^{a)}Author to whom correspondence should be addressed: dawid.szarek@pwr.edu.pl

ABSTRACT

This paper proposes an approach for the estimation of a time-varying Hurst exponent to allow accurate identification of multifractional Brownian motion (MFBM). The contribution provides a prescription for how to deal with the MFBM measurement data to solve regression and classification problems. Theoretical studies are supplemented with computer simulations and real-world examples. Those prove that the procedure proposed in this paper outperforms the best-in-class algorithm.

Published under an exclusive license by AIP Publishing. <https://doi.org/10.1063/5.0093836>

Time-changing fractional dynamics are identified in signals measured in systems of various nature, e.g., biological, technical, financial, etc. One of the models used to describe this phenomenon is the multifractional Brownian motion (MFBM), considered a generalization of the classical fractional Brownian motion (FBM). The MFBM is characterized by a time-dependent Hurst exponent $[H(t)]$, in contrast to the FBM where H is constant. In this article, an artificial intelligence-based method dedicated to the estimation of time-varying $H(t)$ from experimental data has been introduced. The algorithm uses a deep convolutional neural network (CNN) with an encoder-decoder architecture, the autocovariance function (ACVF), and the Euclidean distance matrix of the considered process. We demonstrate how the procedure can be applied when noisy and only a limited amount of measurement information is available for estimation. Simulation studies with different time-changing $H(t)$ show that the proposed algorithm enables estimation of the Hurst coefficient with improved accuracy in relation to Coeurjolly's method, currently perceived as the best in class. We propose a discriminating procedure that can be used for the classification of trajectories for cases with constant and time-varying fractional dynamics, especially important for context labeling

of experimental data. The simulation study is supplemented by experimental studies of signals recorded in a hydrogel system. Although the proposed methodology is demonstrated for the MFBM, the designed and validated approach is of universal meaning and can be applied for systems of any nature where the time-changing fractional dynamics govern a temporal evolution.

I. INTRODUCTION

Complexity identified in measured signals reflects the complex interrelations (structural and functional) valid for observed systems.^{1,2} This property does not depend on the scale and nature of the system; i.e., one can observe complex behavior in geological, meteorological, financial, biological, and engineering objects perceived in a macro-world context, e.g., Refs. ^{3–8}. Nevertheless, the complex patterns are also visible when we characterize the micro-to-nano world, e.g., plasma processes, genomics, and materials science.^{9–11} Fractional calculus is one of the mathematical approaches used to model this class of systems and processes, but applications in measurement science and engineering benefit from simple and efficient procedures for the depiction of the rules

and properties governing complexity. Fractional Brownian motion (FBM) with Hurst exponent H is a stochastic process that can entail long-ranged temporal correlations^{12–15} and has been broadly employed across multiple disciplines.^{16,17} It has been established experimentally and defined theoretically to depict complex interrelations and system dynamics.^{18,19} Typically, the estimation of the constant H on a given scale or within a multiscale view is performed to reveal and label the system state, e.g., healthy vs asthmatic.^{4,20} However, real systems often evolve in time such that the Hurst index is not constant along the whole time series. For example, persistent or anti-persistent processes, where the state changes with time, have been observed in systems as diverse as the motion of proteins in living cells,^{21,22} nanoscale particles in the cytoplasm,^{23,24} and the degradation of Li-ion batteries.²⁵ It has been shown that the index H can change with time, and thus, a more advanced description [with a reconstructed variation in $H(t)$] should be applied. An example of such a process is the generalization of FBM and the multifractional Brownian motion (MFBM),^{26–29} as a theoretical model, which is valid for the description of real-world systems of any nature and across various scales. A mathematical consequence of relaxing the stationarity constraint in the MFBM model lies in the ability to control the local irregularity at the output, which is valid for physical systems. Thus, the MFBM process provides a useful description for a host of continuous and non-stationary natural signals, exhibiting physical complexity encoded in observed systems and, in turn, broadening the field of its application. One can imagine a pointwise irregularity evoked by erosion phenomena in pricing, mountain rock, glacier degradation, etc.^{30–34} Some of these components cannot be directly measured, and they need to be reconstructed from experimental data; e.g., a projection of the relation between cell's configuration and performance in a wireless communication network needs to assume, among other things, time-varying environmental changes, which profile the output of a technical system operating in a given geographical location.³⁵ Furthermore, a challenge in signal processing is to reliably estimate a varying $H(t)$ as the fingerprint of the dynamics when the system is corrupted by a noisy environment using a limited amount of *a priori* information. In this article, a prescription for solving this challenge is provided for the MFBM, considered the classical model with time-changing fractional dynamics. In contrast to FBM, the MFBM assumes a time-dependent Hurst exponent. In the literature, one can find various algorithms for the estimation of a time-dependent Hurst exponent.^{31,36–41}

We develop an approach based on artificial intelligence (AI) that is verified with numerical simulations and a real-world system: signals recorded in a microscale experiment with polystyrene nanobeads placed in an agarose hydrogel. The proposed technique utilizes deep convolutional neural networks (CNNs) with encoder–decoder architecture.^{42–44} The presented results, i.e., the method of time-varying $H(t)$ estimation, and qualitative and quantitative observations performed for a theoretical and real-world case can be translated to the real-world system/process of any nature and scale where the fractional dynamics can be explained by the time-changing Hurst exponent.

The application of the AI and machine learning (ML) techniques to signal processing brings vivid interest of researchers and engineers^{45,46} since this approach is efficient to the exploration of

complex problems; still, the open challenge for an AI/ML domain is concluded with a short *a priori* knowledge available.⁴⁷ Used in the article, the artificial neural networks (ANNs) exhibit additional advantages; e.g., when trained, a one-step conclusion is realized in the ANN for a given input data, which is of significant meaning for real-time applications, where an iterative scheme of the adaptive procedures can be ineffective; this is the case for a parameter estimation of the time-dependent processes (e.g., MFBM). On the other hand, the challenge is a relatively high expectation of the ANN as regards the amount of data used during training. This triggers fundamental and applied studies in the range of ANN, measurement procedures, and signal processing focused on the development of the methods and tools valid for reliable characterization of complex systems and signals with a limited amount of corrupted data.^{48,49} The problem identified in the paper and the prescription developed for solving it bring contributions both in a range of the studied process (MFBM) and tools (ANN applied to modeling of the fractional dynamics with a limited amount of input data). A recent signal processing literature indicates this research area as very promising, also in the area of fractional dynamics systems.^{50–53}

The other contribution introduced in the paper is a discriminating procedure that can be applied to sort out the trajectories into two classes: classical fractional dynamics systems (the case with a constant Hurst index) and time-dependent fractional dynamics (the case with a time-dependent Hurst exponent). As provided, in the paper, the methodology for intelligent labeling of noisy data with a short history opens new opportunities for signal processing, e.g., toward real-time algorithms for annotation of time series data, where the stated existence of the FBM and MFBM components can indicate time slots with various modes of system operation. This method enables a further practical use of the contribution, e.g., to automatize the monitoring of physiological systems (telemedicine), identification of normal and critical events among the meteorological data, and in industrial applications of IoT (Industry 4.0).

The rest of the paper is organized as follows: In Sec. II, we review the most important properties of the FBM and MFBM. In Sec. III, we introduce the estimation algorithm with the description of the proposed model's architecture alongside its training methodology and the random dataset construction. Next, in Sec. IV, we check the effectiveness of the proposed technique for the MFBM time-varying output with different Hurst exponent functions. The efficiency of the ANN-based technique is compared with the results of the approach proposed in Ref. 39. In this section, we also propose a step-by-step procedure for discriminating the cases with constant and time-dependent fractional dynamics, characterized by the Hurst index. In Sec. V, we perform the analysis for the experimental data. Section VI summarizes the paper.

II. FRACTIONAL AND MULTIFRACTIONAL BROWNIAN MOTION

Fractional Brownian motion can be considered an extension of classical Brownian motion.^{14,15} FBM $\{X_H(t), t \geq 0\}$ with the Hurst exponent (also called the Hurst index) $H \in (0, 1)$ is a zero-mean Gaussian process that fulfills the following Langevin equation:^{12,54}

$$\frac{dX_H(t)}{dt} = D\eta_H(t). \quad (1)$$

In the above equation, $\{\eta_H(t), t \geq 0\}$ is the discrete fractional Gaussian noise (FGN), which is a zero-mean Gaussian process with the following autocovariance function (ACVF):

$$\mathbb{E}[\eta_H(0)\eta_H(t)] = D((t+1)^{2H} + (t-1)^{2H} - 2t^{2H}), \quad t \geq 0. \quad (2)$$

The parameter $D > 0$ is the so-called diffusion coefficient. It is worth highlighting that the FGN is a stationary process. The process could also be defined using fractional integration^{27,37,55} with the Riemann–Liouville fractional derivative,⁵⁶ which is a special case of the Dzherbashian–Nersesian fractional operator.⁵⁷

The ACVF of FBM is given by

$$\mathbb{E}[X_H(t)X_H(s)] = D(t^{2H} + s^{2H} - |t-s|^{2H}), \quad t, s \geq 0. \quad (3)$$

For $H > 1/2$, the FGN is a positively correlated process and exhibits long-range dependence (long-memory or persistence).¹² In this case, the FBM is considered a super-diffusive process. For $H < 1/2$, the FGN is a negatively correlated process, and it exhibits short-range dependence (medium dependence or anti-persistence). In this case, FBM is a sub-diffusive process. For $H = 1/2$, the FBM reduces to the standard Brownian motion (BM) $\{X_{1/2}(t), t \geq 0\}$.

The multifractional Brownian motion $\{X_{H(t)}(t), t > 0\}$ is one of the extensions of the classical FBM defined above. The MFBM is a zero-mean Gaussian process with the following ACVF for $t, s \geq 0$:^{26,58}

$$\begin{aligned} \mathbb{E}[X_{H(t)}(t)X_{H(s)}(s)] \\ = D(H(t), H(s)) (t^{H(t)+H(s)} + s^{H(t)+H(s)} - |t-s|^{H(t)+H(s)}), \end{aligned} \quad (4)$$

where $D(\cdot, \cdot)$ is the function given by

$$D(x, y) = \frac{\sigma^2 \sqrt{\Gamma(2x+1)\Gamma(2y+1)} \sin(\pi x) \sin(\pi y)}{2\Gamma(x+y+1) \sin(\pi \frac{x+y}{2})}. \quad (5)$$

In the above equations, $\sigma > 0$ is a constant, $\Gamma(\cdot)$ is a Gamma function, and $H(\cdot) : [0, \infty) \rightarrow [a, b] \subset (0, 1)$ is a Hölder function.

Similar to the classical FBM, the function $H(\cdot)$ is also called the Hurst exponent. For interesting properties of MFBM, see, for instance, Ref. 37.

In this paper, we consider the following Hurst exponent $H(\cdot)$ defined for $t \in [0, T]$ ($T > 0$ is a time horizon)

- Case 1—constant Hurst exponent:

$$H(t) \equiv H; \quad (6)$$

- Case 2—linear Hurst exponent:

$$H(t) = a_1 t + a_2; \quad (7)$$

- Case 3—logistic Hurst exponent:

$$H(t) = b_1 + \frac{b_2}{1 + \exp\{-b_3(\frac{t}{T} - b_4)\}}; \quad (8)$$

- Case 4—periodic Hurst exponent:

$$H(t) = c_1 \sin\left(c_2 \frac{t}{T}\right) + c_3. \quad (9)$$

As it was mentioned, case 1 corresponds to the classical FBM. Case 2 corresponds to a process that switches steadily from sub- to super-diffusive regimes (or vice versa). In case 3, we have a similar situation; however, the change takes place more rapidly, which can be used to model a fast transition in the mode of motion.^{59,60} The Hurst exponent considered in case 3 approximates the regime switching Hurst exponent. More precisely, with $b_1 = H_1$, $b_2 = H_1 - H_2$, $b_4 = \tilde{T} \in (0, T)$, we have

$$H(t) = H_1 + \frac{H_2 - H_1}{1 + \exp\{-b_3(\frac{t}{T} - \tilde{T})\}} \stackrel{b_3 \gg 0}{\approx} \begin{cases} H_1, & t < \tilde{T}; \\ H_2, & t \geq \tilde{T}. \end{cases} \quad (10)$$

Specifically, the corresponding process approximates a process that switches at time $\tilde{T}$ from a FBM with Hurst exponent $H = H_1$ to a FBM with $H = H_2$. A particular case is when $H_1 < 0.5 < H_2$. This is the situation where the process for times $t < \tilde{T}$ exhibits sub-diffusive behavior, while for $t > \tilde{T}$, it is a super-diffusive one. Case 4 corresponds to the situation where the regime changes are gradual and repetitive.

III. ESTIMATION ALGORITHM

The estimation of the Hurst exponent $H(t)$ for the MFBM based on a single trajectory is a complex problem. The main rationale behind this statement is that we want to estimate $\{H(t)\}_{t=1}^T$ vector of length T given a sample trajectory of length T $\{X_{H(t)}(t)\}_{t=1}^T$. Thus, to estimate the value of the Hurst exponent for a given time t , one can rely solely on a considered trajectory at point t and its surroundings. Depending on how locally ergodic the process is, we can use a smaller or bigger window size to estimate $H(t)$. Regarding the limited amount of *a priori* information of the process properties encoded in the input data and the unclear rules governing the selection of the window size for analysis, the problem is solved in the paper with the use of a deep learning methodology based on convolutional deep neural networks.

To enrich the input information about the process, $M = 4$ different distance measures have been applied to the raw signal during the preprocessing phase, namely,

- $\{\log(1 + |x_i - x_j|)\}_{ij}$,
- $\{\log(1 + (x_i - x_j)^2)\}_{ij}$,
- $\{\log(1 + |x_i x_j|)\}_{ij}$,
- $\{\log(1 + (x_i x_j)^2)\}_{ij}$,

where $i, j \in \{1, 2, \dots, T\}$.

Distances (a) and (b) resemble the Euclidean distance, while the last two metrics are in analogy to the covariance. Since the MFBM can be defined through the covariance matrix (like any Gaussian process), the empirical covariance matrix can be used to characterize a given signal. The motivation to use the Euclidean distance comes from the ACVF's (4) interpretation, i.e., the higher $H(t)$, the higher is the variance of the process and thus raises the probability of increasing the distance between observations. The distance expressed by the function $\log(1 + |x|)$ is used to give more significance to values closer to zero, while the distance based on $\log(1 + x^2)$ is used to emphasize more distant and rapid changes.

The $\log(\cdot)$ function is used to achieve resilience to the outliers and stability of the estimator.

The deep learning model consists of two parts: an encoder and a decoder.⁴⁴ The transformation is made from a three-dimensional ($T \times T \times M$) input into a one-dimensional vector ($T \times 1 = T$) corresponding to the values ($H(1), H(2), \dots, H(T)$). Both the encoder and decoder are based on Inception-ResNet,⁶¹ and the whole network resembles U-Net.⁴⁴ The schematic representation of the structure applied to the model is shown in Fig. 1. Details of each block are described in Appendix B.

The selection of Inception-ResNet, as an inspiration for both the encoder and decoder, was motivated by the characteristics of the inception network.⁶² Specifically, combining both more local (3×3) and wider (7×1 and 1×7) filters enables to capture a variety of dependencies. Furthermore, this architecture implements residual connections, introduced with the ResNet,⁶³ which enables

building significantly deeper NNs trained with a backpropagation technique⁶⁴ and improves convergence during the training process.

In the paper, the estimation problem consists of mapping a tensor ($T \times T \times M$) onto a vector ($T \times 1 = T$). As the result, the network (Fig. 1) was also significantly inspired by the U-Net's architecture.⁴⁴ However, in the classic implementation, the order of the tensor does not change. For this reason, the architecture was modified such that the encoder works on three-dimensional tensors and the decoder on two-dimensional ones. To extract the most important (dominant) feature from each column, a reduction method was applied to the input data where the maximal value over one of the T axes was taken. It is crucial as if the average is taken, and then the information related to the whole signal is used. This stands in contradiction to the locality of the estimates. The network was trained using randomly generated outputs. In this scenario, each time a unique realization of the trajectory is provided to the NN

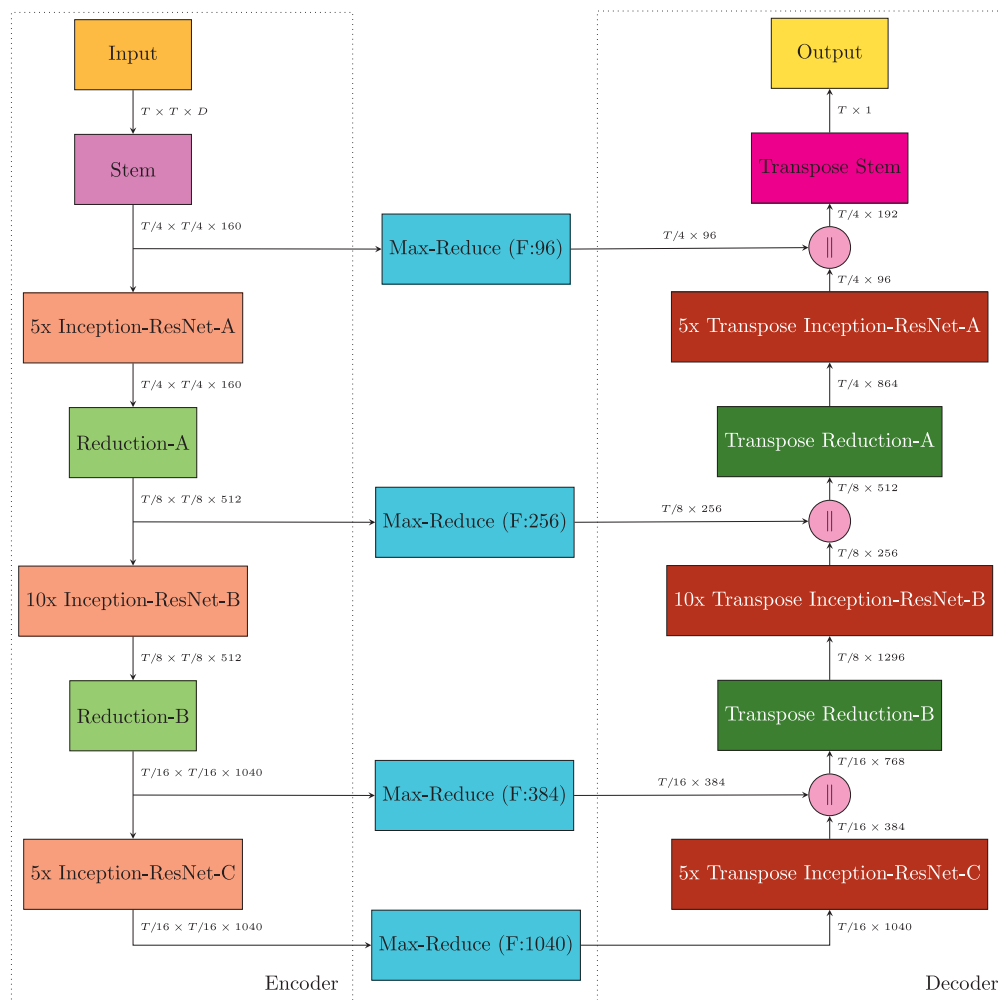


FIG. 1. The architecture of an encoder-decoder structure of NN designed and applied during the experiment.

TABLE I. Parameterization applied to the $H(t)$ function for the following cases of computer experiments.

Case	Parameters' sampling method
Case 1—Eq. (6)	$H \sim \mathcal{U}(0.01, 0.99)$
Case 2—Eq. (7)	$H_0 \sim \mathcal{U}(0.01, 0.99), H_1 \sim \max(\min(H_0 + \mathcal{N}(0, 0.25), 0.99), 0.01), a_1 = \frac{H_1 - H_0}{T}, a_2 = H_0$
Case 3—Eq. (8)	$H_0 \sim \mathcal{U}(0.01, 0.99), H_1 \sim \max(\min(H_0 + \mathcal{N}(0, 0.25), 0.99), 0.01)$ $b_1 = H_0, b_2 = H_1 - H_0, b_3 = \mathcal{U}(10, 250), b_4 = \mathcal{U}(0.1, 0.9)$
Case 4—Eq. (9)	$\bar{H} \sim \mathcal{U}(0.02, 0.98), c_1 \sim \mathcal{U}(0.01, \min(0.99 - \bar{H}, \bar{H})), c_2 = \mathcal{U}(1, 200), c_3 = \bar{H}$
RBM—Appendix A 1	$H_0 \sim \mathcal{U}(0.1, 0.9), \sigma \sim \mathcal{U}(0.01, 0.99)$
ROU—Appendix A 2	$H_0 \sim \mathcal{U}(0.1, 0.9), \sigma \sim \mathcal{U}(0.01, 0.99), \lambda \sim \mathcal{U}(0.1, 100)$
ST—Appendix A 3	$H_0 \sim \mathcal{U}(0.01, 0.99), H_1 \sim \mathcal{U}(0.01, 0.99), \gamma \sim \mathcal{U}(0.1, 10), \beta \sim \mathcal{U}(0.1, 100)$

input. The batch consisted of two samples; thus, the input tensor has the following shape: $2 \times T \times T \times M$ (as previously stated $M = 4$). The T value (corresponding to the number of samples of a trajectory) was randomly selected from $T \in \{256, 384, 512\}$. If it had not been specified in the other way, the evaluations were based on the largest $T = 512$. For a single signal, a random $H(t)$ is selected—one of the functions presented in Eqs. (6)–(9). Moreover, the random Hurst index was added to the learning set, to reconstruct the possible random changes of this parameter, which is especially important during the analysis of experimental data (presented in Sec. V). We include the cases when $H(t)$ is a stochastic process. Here, three processes have been considered, i.e., the reflected Brownian motion (RBM), the reflected Ornstein–Uhlenbeck (ROU) process, and the smoothed telegraph process (ST). The corresponding definitions are presented in Appendix A, and the parameterization for these processes applied during experimentation is specified in Table I.

Each function has parameters that are selected randomly in the training set using random variables defined in Table I. Then, based on a trajectory, M distance matrices are computed and scaled using the following formula:

$$X_{..k}^{\text{scaled}} = \frac{X_{..k}}{\sqrt{\frac{1}{T^2} \sum_{i=1}^T \sum_{j=1}^T X_{ijk}^2}}, \quad k \in \{1, \dots, D\}. \quad (11)$$

Additionally, normalization is applied to the input data:

$$\mu_k = \frac{1}{NT^2} \sum_{n=1}^N \sum_{i=1}^T \sum_{j=1}^T X_{nijk}^{\text{scaled}}, \quad (12)$$

$$\sigma_k = \sqrt{\frac{1}{NT^2} \sum_{n=1}^N \sum_{i=1}^T \sum_{j=1}^T (X_{nijk}^{\text{scaled}})^2 - \mu_k^2}, \quad (13)$$

$$X_{..k}^{\text{norm}} = \frac{X_{..k}^{\text{scaled}} - \mu_k}{\sigma_k}, \quad k \in \{1, \dots, D\}. \quad (14)$$

Parameters μ_k and σ_k are estimated before the training using $N = 1024$ randomly generated samples.

Since the values depend only on themselves, the scaling was applied. This operation enables to control the range of the input regardless of the type of diffusion or $H(t)$. In the study, the first scaling is used to equate the scales of each distance matrix. The input normalization [illustrated as a block of the network in Fig. 13(a)]

assures the zero mean and unit variance for the input; this normalization accelerates and stabilizes the convergence of the optimization algorithms.

The training duration was set to 40 epochs, each consisting of 4048 batches. Several methods to monitor and tune the learning process were applied. First, the learning rate scheduler was used, which contains a warm-up phase. The following recurrent function for learning rate profiling was implemented:

$$lr(\text{epoch}) = \begin{cases} 10^{-6}, & \text{epoch} = 0; \\ 6 \cdot lr(\text{epoch} - 1), & \text{epoch} \in \{1, 2, 3\}; \\ 0.975 \cdot lr(\text{epoch} - 1), & \text{epoch} \geq 4. \end{cases} \quad (15)$$

On top of that, a learning rate reduction in the plateau (after 3 epochs of stagnation by a factor of 0.3) and early stopping⁶⁵ (after 7 epochs) conditions were added to mitigate jumping around an optimum and to stop the computations when no further improvement is observed, respectively. Both use the validation error to make an informed choice whenever the learning rate needs to be reduced or the training stopped. The Adam optimizer was used to fit the model to the data during training (while the batch size caused by hardware limitations is small, we increased parameter $\beta_1 = 0.99$).^{66–71} The mean squared error (MSE) function was used as the loss function during optimization.

IV. COMPUTER VALIDATION OF THE PROPOSED ALGORITHM

In this section, we present the effectiveness of the methodology presented in Sec. III and propose a visual test to discriminate if the real data correspond to the MFBM with a constant or time-dependent Hurst exponent.

The effectiveness of the proposed estimation algorithm is evaluated using 500 simulated signals of length $T = 512$ for cases 1–4 discussed in Sec. II. A time-varying MFBM output is simulated with the use of the algorithm based on the Cholesky decomposition for Gaussian processes.^{72,73} For the validation, we use MFBM with the parameters presented in Table II. The estimation results are presented in Fig. 2. With the light red lines, we present the 95% confidence intervals (calculated based on the obtained estimated Hurst exponent), while with the dark red line, we present the median of $\hat{H}(t)$. The blue line corresponds to the theoretical value of the Hurst exponent.

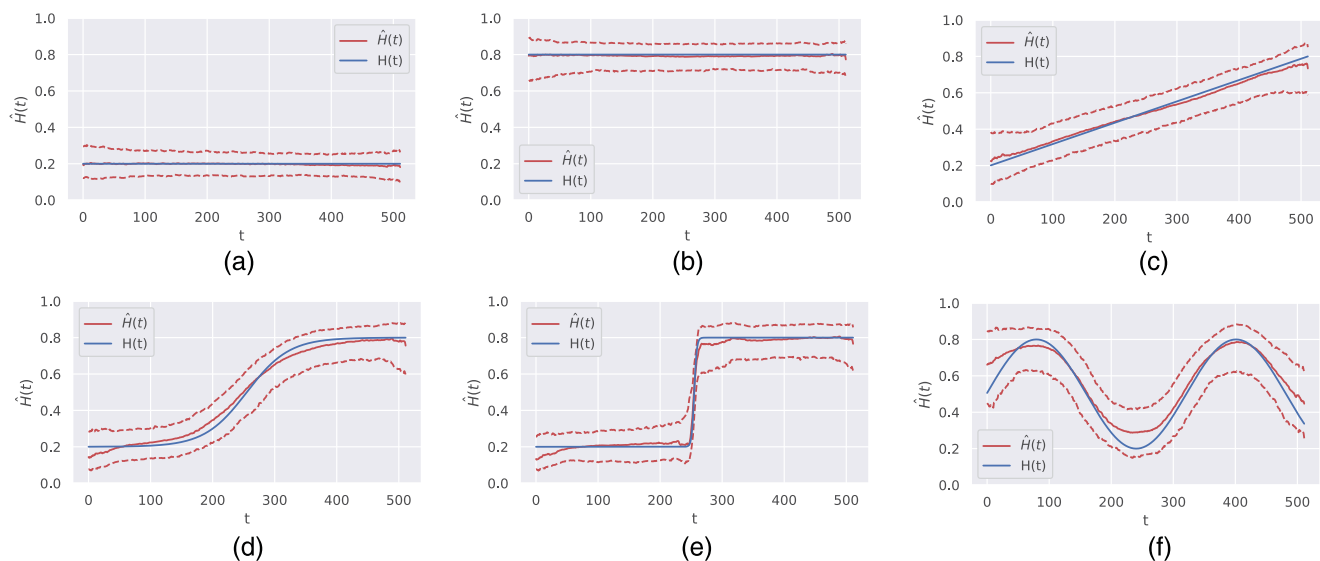
TABLE II. Parameters' values used for each considered case of $H(t)$ function in the computer validation.

Case	Parameters
1(a)	$H = 0.2$
1(b)	$H = 0.8$
2	$a_1 = \frac{0.8-0.2}{1000}, a_2 = 0.2$
3(a)	$b_1 = 0.2, b_2 = 0.6, b_3 = 15, b_4 = 0.5$
3(b)	$b_1 = 0.2, b_2 = 0.6, b_3 = 250, b_4 = 0.5$
4	$c_1 = 0.3, c_2 = 10, c_3 = 0.5$

One can observe that the proposed algorithm estimates $H(t)$ correctly (with wider confidence intervals for small and large times). Both examples—constant functions [Cases 1(a) and 1(b)]—are estimated accurately. The confidence intervals are narrow, and the estimated functions are very close to the theoretical ones. For the linear function (Case 2), one can observe a significant increase of uncertainty at the boundary. This behavior is in line with intuition as there is less evidence that this function continues to be linear outside the boundary. By interpreting the confidence intervals, the model prefers a constant solution at the boundaries. In the following examples, one can observe that the more volatile $H(t)$, the more unstable the estimator $\hat{H}(t)$. This can be observed by comparing the twin examples—Cases 3(a) and 3(b). With smoother $H(t)$ [Case 3(a)], the quantile lines of $\hat{H}(t)$ are also smoother and less variable compared to Case 3(b), where one can observe sudden jumps and distortions of the estimates. In the situation when $H(t)$ is periodic (represented by Case 4), the estimator underestimates and favors the solution closer to a constant, i.e., the mean value.

We have compared the obtained results using the proposed algorithm with the technique considering the classical method for estimation of the Hurst exponent for MFBM presented in Ref. 39. This method is based on the observation that the convolved trajectory (with a specific wavelet filter) produces a linear relation with the dilatation rate of the filter. In this relation, the slope is proportional to $H(t)$ at a given time t , and it can be extracted using basic transformations. However, it cannot be done for all $t \in \{0, \dots, T\}$ because the filter (wavelet) length amplified by the dilatation limits the number of beginning times t for which the estimate can be evaluated. We have made the comparison using two lengths of signals: $T = 256$ and $T = 2560$, with the examples proposed in Ref. 39 [with proposed optimal parameters for each of the $H(t)$ functions]. One can observe (see Fig. 3 and the summary statistics in Table III) that for $T = 256$, the proposed estimation algorithm outperforms the classical technique. First of all, the proposed method allows one to estimate $H(t)$ for all values of the time points $t \in \{1, 2, \dots, T\}$. This is crucial, especially for small trajectories, because, with such a limitation, we can lose information about a large percentage of a trajectory as the estimate does not exist. The algorithm presented in Ref. 39 (with proposed optimal parameters) is unable to estimate the first 39 values of $\{H(t)\}_{t=1}^T$, effectively producing a vector $\{\hat{H}(t)\}_{t=40}^T$.

The comparative studies indicate that the variance of the estimator presented in Ref. 39 is higher than the variance of the introduced estimator. For the case presented in Ref. 39, namely, for $T = 2560$, the considered methods coincide with the introduced algorithm having a slightly lower variance. However, it has bias issues on the boundaries. This is mainly caused by the fact that the model was not trained on such long trajectories, and it has learned an estimator $\hat{H}(t)$ for which $\forall_{t < 0} H(t) = H(0) \wedge H'(t) = 0$ and $\forall_{t > T} H(t) = H(T) \wedge H'(t) = 0$. Despite this, in general, the

**FIG. 2.** Quantile lines of $\hat{H}(t)$ for each considered case overlaid on true $H(t)$. The dark red lines represent the median and the lighter are 95% confidence intervals. We used 500 Monte Carlo simulations of the trajectories of the length $T = 512$.

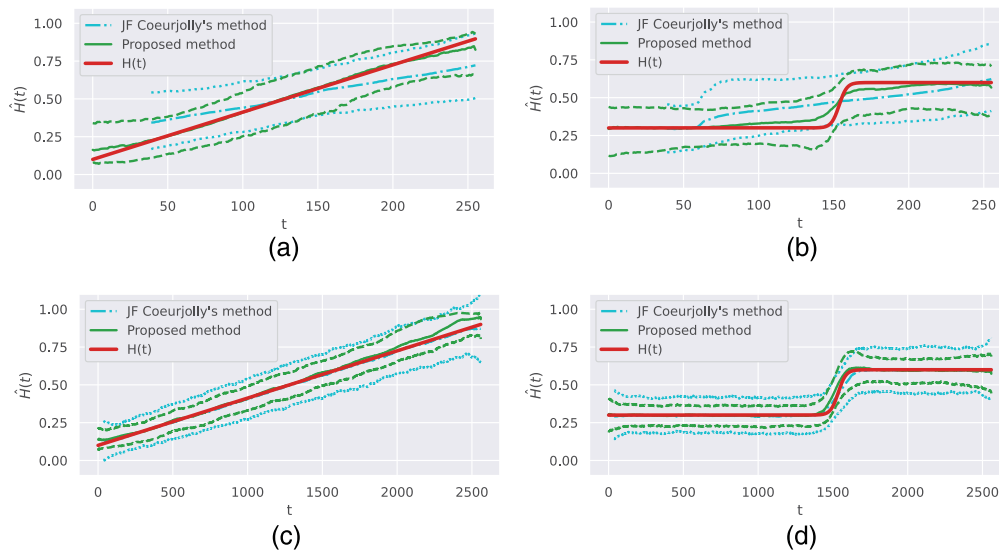


FIG. 3. Comparison of the results obtained using the introduced algorithm and the technique proposed in Ref. 39. The quantile lines of $\hat{H}(t)$ are drawn to depict the estimators' characteristics for cases considered in Ref. 39. The middle line depicts the estimators' median, while the surrounding lines correspond to 95% confidence intervals.

proposed estimator characterizes the lower error (see Table III with summary statistics); thus, one can conclude that the algorithm presented in this paper is more effective overall, especially, for shorter trajectories.

It is worth highlighting that the algorithm proposed in this paper is superior with respect to the algorithm described in Ref. 39 because it does not need to find optimal hyperparameters. The model is equipped with optimal filters and aggregates the information with an optimal window size, whereas in the method proposed in Ref. 39, it is needed to specify appropriate wavelets, with an optimal aggregation window, where choice has a tremendous effect on the estimate. Choosing a too wide window can lower the variance of the estimator; however, it will increase the bias toward a constant solution. With a too narrow window, the estimator can capture rapid changes in the process's characteristics; yet, because of the increased variance, it might be random and this could lead

to incorrect conclusions. This is difficult, especially because, as it is shown in Ref. 39, the optimal hyperparameters are different for cases with the linear and logistic $H(t)$. This yields that there is a need to know what type of $H(t)$ function the process is governed by before the estimation algorithm starts. Using neural networks, we opt for an optimal solution without the need for guessing the optimal parameter and knowing what type of function $H(t)$ is considered. It is because we train the model with several types of functions and because the convolutional neural networks' feature is that they have shared filters' weights. This guarantees for the method to perform well even if a function does not strictly belong to the family of functions the model was trained on. For this reason, the proposed solution will perform better not only for functions originating from training set (see comparison for exemplary functions, with parameterization presented in Table II, shown in Fig. 4 supplemented with summary statistics presented in Table IV) but also for any piecewise-defined function. Furthermore, because we can approximate any function by a piecewise linear function, thus, this method is universal and versatile. We demonstrate it on the example shown in Fig. 5 in which a similar simulation approach is used as in Fig. 3 but with a trajectory length $T = 512$ and a more complex $H(t)$ function—the piecewise constant and the sinusoidal function. One can observe that the method proposed in this paper indeed captures the temporal rapid changes of $H(t)$ even though such a function is not in the set of training functions—only its components. Furthermore, from the comparison with the algorithm proposed in Ref. 39, the new method allows one to discover a much more convoluted $H(t)$, whereas the other one smooths out this region causing the temporal periodic behavior to be impossible to deduce. This performance difference is even more emphasized in Table V, where the error metrics for Coeurjolly's method³⁹ are almost three times larger.

TABLE III. Error summary metrics for tested methods and cases presented in Fig. 3. We indicate as Method A the proposed method and as Method B Coeurjolly's method.

Case	T	Method	MAE	RMSE
Linear case	256	A	0.0566	0.0732
		B	0.0971	0.1237
	2560	A	0.0380	0.0485
		B	0.0560	0.0710
Logistic case	256	A	0.0629	0.0800
		B	0.1052	0.1293
	2560	A	0.0340	0.0443
		B	0.0549	0.0691

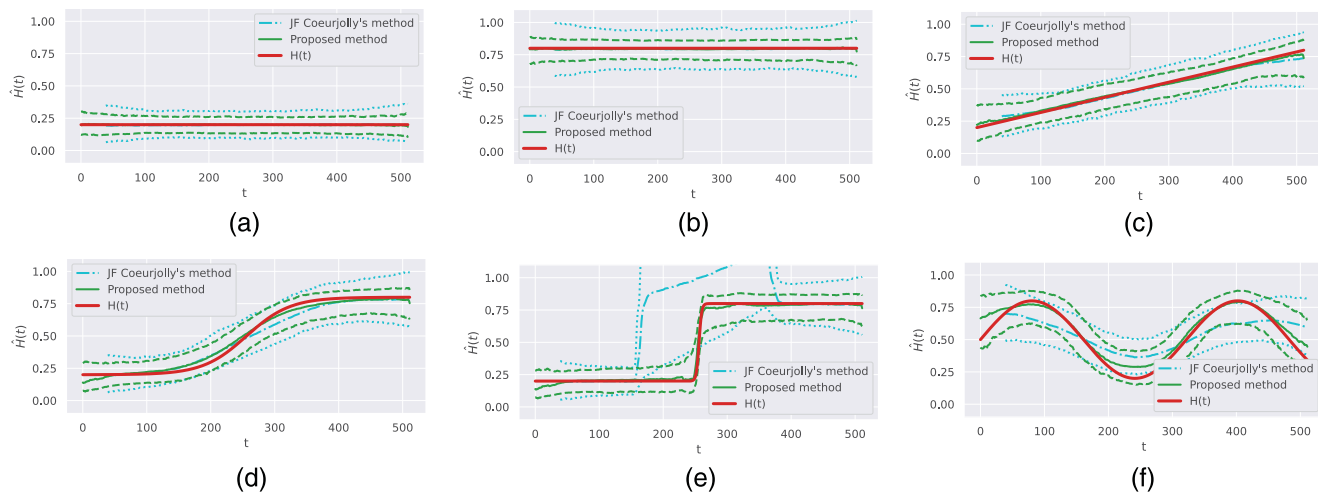


FIG. 4. Comparison of the results obtained using the introduced algorithm and the technique proposed in Ref. 39. The quantile lines of $\hat{H}(t)$ are drawn to depict the estimators' characteristics for cases whose error metrics are summarized in Table IV. The middle line depicts the estimators' median, while the surrounding lines correspond to 95% confidence intervals. To compute those, we performed 500 Monte Carlo simulations with trajectories of the length $T = 512$.

A. Discriminating algorithm between cases with a constant and time-varying Hurst exponent

As it was shown in the above analysis, even if the analyzed data correspond to the MFBM with a constant Hurst exponent (i.e., FBM), the estimated $H(t)$ is not constant. Furthermore, it varies around the true value of the Hurst exponent, which comes from the fact that the calculated $\hat{H}(t)$ from the data is only an estimator that, from a mathematical point of view, is a random variable. Hence, there is a need to discriminate the cases when the non-constant $\hat{H}(t)$ corresponds to the MFBM with the Hurst exponent being a time-varying deterministic function and FBM [$H(t)$ is constant]. Consequently, in this part, we propose a simple visual test that

can be used to discriminate the mentioned cases. The discriminating algorithm proceeds as follows:

- For a real signal of length T , estimate the Hurst exponent $\hat{H}_0(t)$, $t = 1, 2, \dots, T$ using the proposed estimation algorithm;
- Calculate the mean of the obtained values $\hat{H}_0(t)$ along t . This value we denote as H_{test} ;
- Simulate M FBM time-invariant outputs with the Hurst exponent H_{test} of the length T ;
- For each simulated signal, estimate the Hurst exponent using the proposed estimation algorithm. As a consequence, we obtain M values of $\hat{H}(t)$;

TABLE IV. Error summary metrics for the cases from Sec. II with parameters defined in Table II.

Case	Method	MAE	RMSE
1(a)	Proposed method	0.0277	0.0350
	Coeurjolly's method	0.0460	0.0578
1(b)	Proposed method	0.0331	0.0420
	Coeurjolly's method	0.0681	0.0855
2	Proposed method	0.0429	0.0551
	Coeurjolly's method	0.0621	0.0793
3(a)	Proposed method	0.0465	0.0594
	Coeurjolly's method	0.0684	0.0872
3(b)	Proposed method	0.0436	0.0576
	Coeurjolly's method	0.2679	0.4248
4	Proposed method	0.0693	0.0889
	Coeurjolly's method	0.1231	0.1475

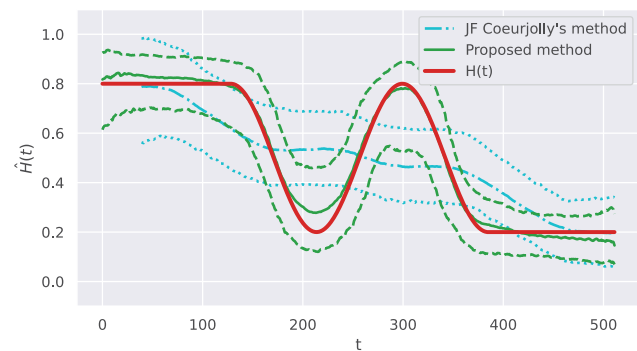


FIG. 5. Comparison of the results obtained using the introduced algorithm and the technique proposed in Ref. 39. The quantile lines of $\hat{H}(t)$ are drawn to depict the estimators' characteristics when $H(t)$ is a piecewise constant and sinusoidal function. The middle line depicts the estimators' median, while the surrounding lines correspond to 95% confidence intervals. Case with $T = 512$.

TABLE V. Error summary metrics for example from Fig. 5.

Method	MAE	RMSE
Proposed method	0.0584	0.0771
Coerjolly's method	0.1492	0.1860

- For each $t = 1, 2, \dots, T$, calculate the empirical quantiles on the level $\alpha/2$ and $1 - \alpha/2$ and construct the empirical confidence interval for the Hurst exponent on the level $1 - \alpha$. For a given t , we denote it as $[Q_{\alpha/2}(t), Q_{1-\alpha/2}(t)]$;
- For each $t = 1, 2, \dots, T$ check if $\hat{H}_0(t)$ falls into the constructed confidence interval. If for minimum $100(1 - \alpha)\%$ of the t values it is satisfied that $\hat{H}_0(t) \in [Q_{\alpha/2}(t), Q_{1-\alpha/2}(t)]$, then we can suspect that the signal corresponds to the FBM. Otherwise, we cannot assume the constant Hurst exponent and assume the signal corresponds to the MFBM with time-varying $H(t)$.

The discrimination between constant and time-varying Hurst exponents can be performed based on the plot (visual inspection), where we demonstrate the functions $\hat{H}_0(t)$ and the constructed empirical confidence intervals $[Q_{\alpha/2}(t), Q_{1-\alpha/2}(t)]$ along $t = 1, 2, \dots, T$ or on the basis of the percentage of time points for which the estimated Hurst exponent falls into the constructed confidence intervals.

To demonstrate the effectiveness of the proposed discriminating algorithm, first, we analyze the MFBM time-varying output with the linear Hurst exponent corresponding to Case 2, see Eq. (7), and the same parameters used in the simulation study (listed in Table II). Here, we consider the signals of length $T = 512$. For the simulated signal, we perform the steps mentioned above, and in Fig. 6(a), we show the estimated Hurst exponent and the constructed confidence interval (and the median) for $\alpha = 0.05$. We assume $M = 1000$ as the number of FBM time-invariant outputs used to construct the confidence intervals and the median. As one can see, the estimated Hurst exponent does not fall into the constructed confidence interval for most of the t values (81%) and the hypothesis of the constant $H(t)$ is rejected.

As a second example, we analyze the FBM time-invariant output with the Hurst index equal to the mean of $H(t)$ used in the linear case described above with the value equal to $H = 0.5$. In Fig. 6(b), we demonstrate the visual test described above for $\alpha = 0.05$ and $M = 1000$. This result indicates that the hypothesis of the constant Hurst exponent cannot be rejected. The percentage of time points for which the estimated $H_0(t)$ falls into the constructed confidence interval is equal to 100%. In this plot, one can also observe that the median obtained from Monte Carlo simulations is not very close to the constant $H(t) \equiv H = 0.5$ from which the process was simulated. The reason for that is the time average taken from the series of estimators $\{\hat{H}(t)\}$ resulted in a value only close to the median. This behavior is strictly connected to ergodic theory and the fact that the following limit $\lim_{T \rightarrow \infty} \frac{1}{T} \sum_{t=1}^T \hat{H}(t)$ does not converge to H as fast as $\lim_{M \rightarrow \infty} \frac{1}{M} \sum_{i=1}^M \hat{H}_i(t) \sim \mathbb{E}\hat{H}(t)$ due to the inter-dependencies in $\{\hat{H}(t)\}$ as all of the estimates $\{\hat{H}(t)\}_{t=1}^T$ use common trajectory, whereas $\{\hat{H}_i(t)\}_{i=1}^M$ are evaluated using M independent trajectories.

V. EXPERIMENTAL DATA ANALYSIS

To evaluate the performance of the proposed algorithm on experimental data, we analyze the trajectories of polystyrene beads of 50 nm in diameter in an agarose hydrogel. The experimental system has been previously described.¹⁶ A 1.5% agarose gel was prepared from agarose powder by dissolving it in phosphate-buffered saline. The polystyrene beads were first heated to 60 °C in 0.5% Tween 20 and introduced into the agarose solution that was maintained at the same temperature and mixed for 15 min. Then, it was transferred to a hot cover-slip where it slowly cooled until it reached room temperature. The beads were imaged in an inverted microscope with a 40× objective at 71 frames/s. Tracking of the individual beads was performed in LABVIEW using a cross-correlation-based algorithm.⁷⁴ A total of 40 trajectories, consisting of 512 data points each, were analyzed here.

It was previously established that the motion of microspheres in agarose hydrogels displays mean squared displacement, power spectral density,¹⁶ and an empirical anomaly measure⁷⁵ consistent



FIG. 6. The visual test for validating if an estimate corresponds to a constant case. The confidence intervals correspond to an average of $\hat{H}(t)$. The percentage on top of the figures represents the percentage of the $\hat{H}(t)$ being out of CI. Case $T = 512$.

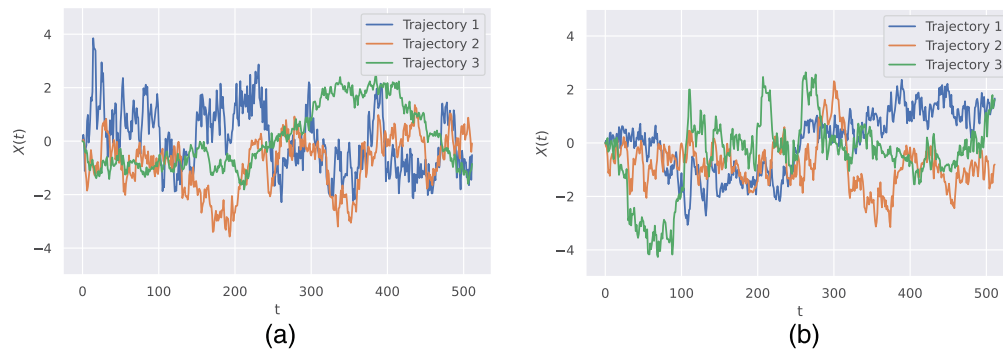


FIG. 7. Sample trajectories of experimental data. (a) Three exemplary trajectories for which we cannot reject the FBM hypothesis with significance level $\alpha = 0.05$. (b) Three exemplary trajectories for which we reject the FBM hypothesis with significance level $\alpha = 0.05$.

with those of FBM. However, the hypothesis that the Hurst exponent is constant in time has never been rigorously tested. Thus, the MFBM testing algorithm provides a good tool for the validation or rejection of FBM.

Representative trajectories are shown in Fig. 7. To demonstrate how the proposed estimation procedure and the discriminating

algorithm described in Sec. IV A can be applied to real data, we demonstrate the estimated $H(t)$ and the constructed confidence intervals calculated for the trajectories of FBM with the Hurst exponent equal to the mean of $\hat{H}(t)$ (Fig. 8), similar to what was described in the discriminating algorithm. In the upper panels, we demonstrate the results for the trajectories presented in Fig. 7, for which

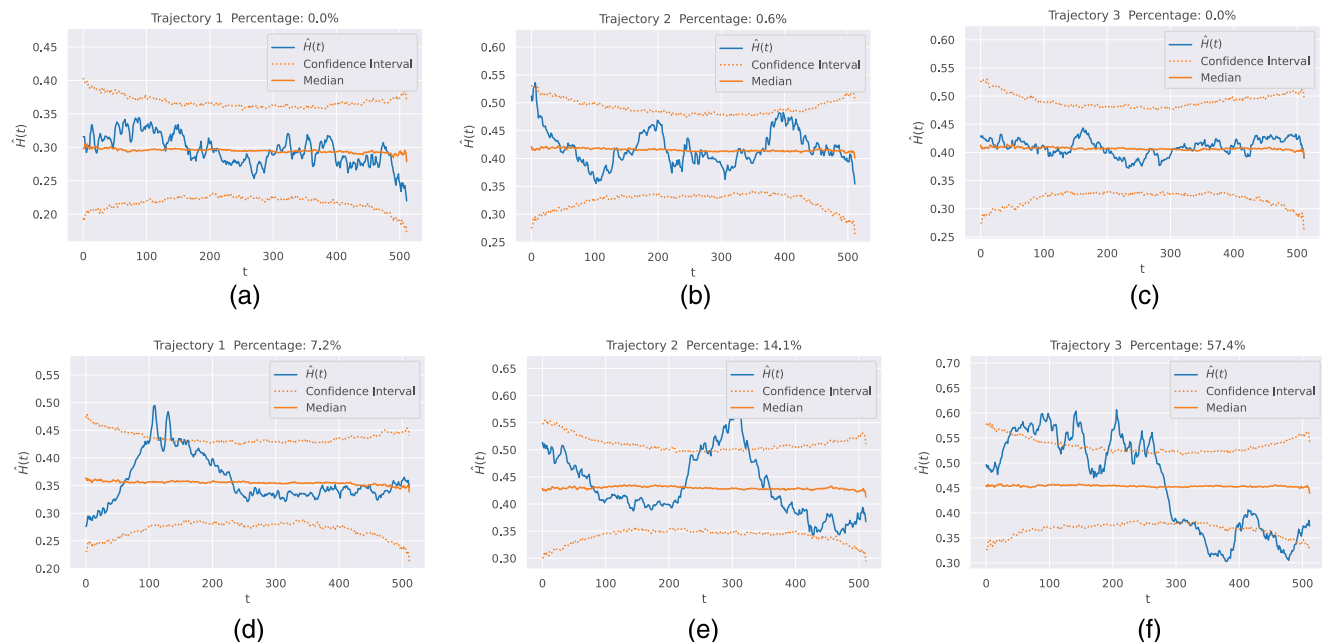


FIG. 8. The visual test for validating if a given trajectory corresponds to FBM [constant $H(t)$]. The confidence intervals are constructed by FBM time-invariant outputs with the Hurst exponent equal to the mean of $\hat{H}(t)$, $t = 1, 2, \dots, T$. On the top, we present the percentage of $\hat{H}_0(t)$ out of the confidence intervals. The trajectories in the upper panels were found with the percentage below the set $\alpha = 0.05$ (the hypothesis of FBM cannot be rejected), and in the bottom panels, we demonstrate the exemplary trajectories for which the hypothesis of FBM is rejected (the percentage is higher than $\alpha = 0.05$). (a) First exemplary trajectory for which the hypothesis of FBM cannot be rejected. (b) Second exemplary trajectory for which the hypothesis of FBM cannot be rejected. (c) Third exemplary trajectory for which the hypothesis of FBM cannot be rejected. (d) First exemplary trajectory for which we reject the hypothesis of FBM. (e) Second exemplary trajectory for which we reject the hypothesis of FBM. (f) Third exemplary trajectory for which we reject the hypothesis of FBM.

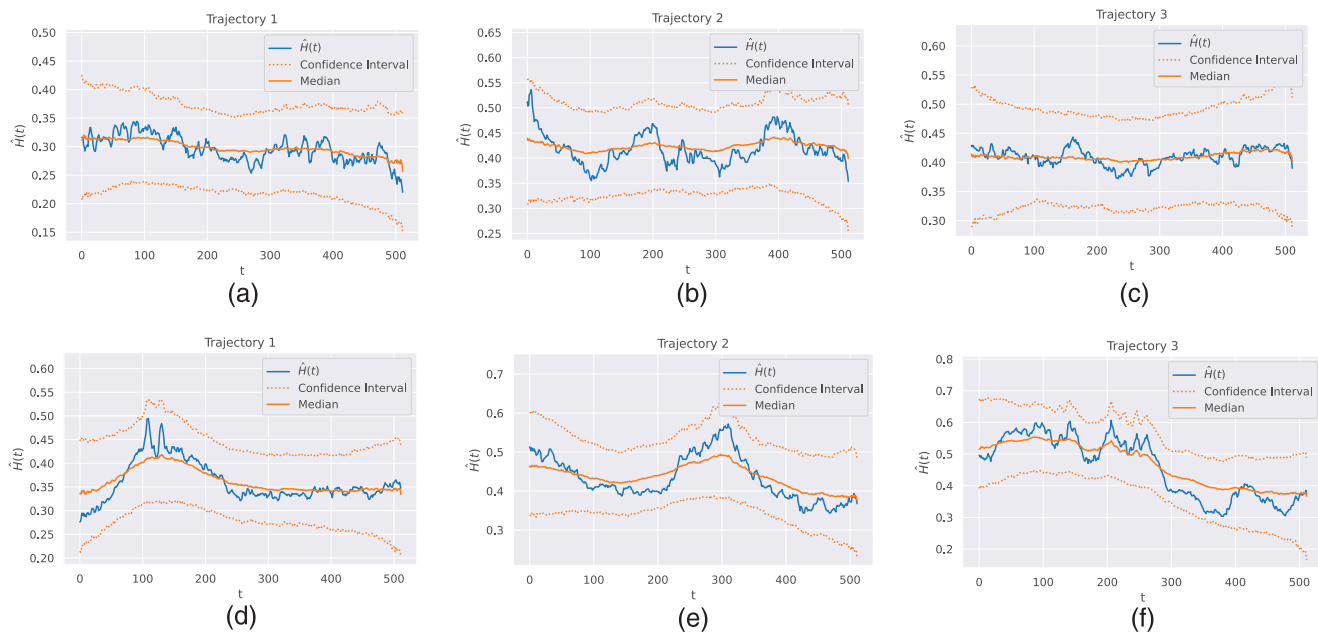


FIG. 9. The estimated Hurst exponents $H(t)$ with their confidence intervals constructed by simulating 1000 trajectories of the MFBM with $\hat{H}(t)$. (a) First exemplary trajectory—constant Hurst exponent. (b) Second exemplary trajectory—constant Hurst exponent. (c) Third exemplary trajectory—constant Hurst exponent. (d) First exemplary trajectory—time-varying Hurst exponent. (e) Second exemplary trajectory—time-varying Hurst exponent. (f) Third exemplary trajectory—time-varying Hurst exponent.

the estimated Hurst exponent falls into the constructed confidence interval for minimum $100(1 - \alpha)\%$ time points (see information on the top of the plots), while in the bottom panels, we show the results for selected trajectories for which we have the opposite situation. In the second case, the hypothesis of FBM cannot be accepted, and we assume the MFBM with the time-varying Hurst exponent. Additionally, in Fig. 8, we also demonstrate the median of the estimated Hurst exponent along time points. In the analysis, we take $\alpha = 0.05$ and $M = 1000$. In general, the number of trajectories for which the hypothesis of FBM cannot be rejected using the introduced discriminating algorithm is equal to 29, while for 11 trajectories, the hypothesis of FBM with constant H is rejected.

In Fig. 9, for trajectories presented in Fig. 7, we show $\hat{H}(t)$ with the confidence intervals constructed using 1000 MFBM time-varying outputs with the estimated Hurst index. One can clearly see that the signals simulated with the corresponding models (see confidence bounds) fully reflect the behavior of the estimated Hurst index, such as rapid changes and the time-varying Hurst index for the cases demonstrated in the upper panels of Fig. 9. The cases presented in the upper panels, according to the discriminating algorithm, correspond to the FBM. Indeed, the estimated Hurst indexes as well as the confidence bounds obtained from the simulated signals exhibit small volatility with respect to the cases demonstrated in the bottom panels. Additionally, it should be highlighted that the estimated $H_0(t)$ (corresponding to the analyzed trajectories) falls into the constricted confidence intervals. From the above analysis, we can conclude that the tested MFBM models are appropriate for the examined trajectories.

The current results provide a validation for the FBM model in 73% of the analyzed trajectories. However, the classical FBM is rejected in the remaining 27%. The average of Hurst indexes indicates that the trajectories are of the sub-diffusion type, as expected from previous analyses. The deviations from a constant Hurst index are usually observed during specific periods at which the trajectories either become super-diffusive ($H > 0.5$) or are seen to exhibit a more marked sub-diffusive behavior. In the former cases, a plausible explanation is that small transient water flows can cause the beads to exhibit directed motion such that the motion appears super-diffusive during these short times. On the other hand, the case of a reduced Hurst index can be caused by imperfections in the hydrogel leading to material heterogeneities. In particular, as a particle traverses a denser region within the medium, it would experience more negative increment autocorrelations and it would transiently appear as FBM with smaller H .

While the majority of the trajectories are found to be of the FBM type, validating previous assumptions, the use of the algorithm for the identification of MFBM in individual trajectories provides very useful information along two different lines of data analysis. First, the method can be used to discriminate trajectories that do not correspond to the simple FBM with constant H . Furthermore, the evaluation of the deviations from constant H provides a tool for the characterization and understanding of these anomalous events. Second, given that a viscoelastic material with homogeneous physical properties should yield trajectories with constant H , the detection of MFBM can be used to evaluate the properties and quality of the material itself. This is, in fact, one of the main goals of microrheology

experiments, where the motions of embedded particles are analyzed in order to measure the mechanical properties of the medium on microscopic scales.⁷⁶

VI. SUMMARY

The contributions provided in the paper can be listed in several highlights:

- The problem of identification and characterization of the complex fractional dynamics is discussed and solved for limited input information and the exemplary model of MFBM [with time-varying $H(t)$ coefficient, adequate to the temporal mechanisms of the fractional system dynamics].
- The AI-based scheme has been designed for constant and time-dependent $H(t)$ estimation and for the sequential binary annotation of the measured signal.
- The MAE and the RMSE estimation errors, measured during the computational experiments for linear, piecewise, and sinusoidal changes of $H(t)$, did not exceed 6% and 8%, respectively, which is half that of the reference method.
- The methodological prescription outlined qualitatively and quantitatively in the paper can be mapped to complex systems of any nature where the observed output exhibits temporal fractional dynamics.
- The designed AI-based algorithm can inspire and contribute to other real-time signal processing schemes operating with short time series in the presence of noise. This work paves the way for developments in various services, including, e.g., telemedicine, IIoT/Industry 4.0, smart city, etc.

ACKNOWLEDGMENTS

The work of A.W. was supported by the National Center of Science under Opus Grant No. 2020/37/B/HS4/00120 “Market risk model identification and validation using novel statistical, probabilistic, and machine learning tools.” D.K. acknowledges the support from the National Science Foundation under Grant No. 2102832.

AUTHOR DECLARATIONS

Conflict of Interest

The authors have no conflicts to disclose.

Author Contributions

Dawid Szarek: Conceptualization (equal); Formal analysis (equal); Investigation (equal); Methodology (equal); Software (lead); Validation (equal); Visualization (lead); Writing – original draft (equal); Writing – review & editing (equal). **Ireneusz Jabłoński:** Conceptualization (equal); Formal analysis (equal); Investigation (equal); Methodology (equal); Project administration (lead); Supervision (lead); Writing – original draft (equal); Writing – review & editing (equal). **Diego Krapf:** Conceptualization (equal); Formal analysis (equal); Investigation (equal); Resources (equal); Supervision (equal); Writing – original draft (equal); Writing – review & editing (equal). **Agnieszka Wyłomańska:** Conceptualization (equal); Formal analysis (equal); Investigation (equal); Methodology (equal);

Project administration (lead); Supervision (lead); Writing – original draft (equal); Writing – review & editing (equal).

DATA AVAILABILITY

The data that support the findings of this study are available from the corresponding author upon reasonable request.

APPENDIX A: STOCHASTIC PROCESSES CORRESPONDING TO THE HURST EXPONENT USED IN THE LEARNING SET

1. Reflected Brownian motion

Brownian motion $\{H_{BM}(t), t \geq 0\}$ is a stochastic process defined by the following Langevin equation:⁷⁷

$$\frac{dH_{BM}(t)}{dt} = \sqrt{2\sigma^2}\eta_{1/2}(t), \quad (\text{A1})$$

where $\{\eta_{1/2}(t), t \geq 0\}$ is the Gaussian white noise process (i.e., process of uncorrelated random variables with zero-mean and constant variance—here, we assume that it is equal to 1) and $\sigma > 0$. The ACVF for BM is given by

$$\mathbb{E}[H_{BM}(t)H_{BM}(s)] = 2\sigma^2 \min\{t, s\}. \quad (\text{A2})$$

The reflected Brownian motion is a way of keeping the Brownian motion within a given domain. In our case, the domain is given by a $(0, 1)$ interval, and the reflection is performed at any given time when the process exceeds it. In the general case, it can be any given domain.

The process we use for $H(t)$ that differs from the definition (A1) is that the starting point $H_{BM}(0) = H_0$.

2. Reflected Ornstein–Uhlenbeck process

The classical Ornstein–Uhlenbeck process $\{H_{OU}(t), t \geq 0\}$ is a stationary solution of the Langevin equation of the following form:⁷⁸

$$\frac{dH_{OU}(t)}{dt} = \lambda(\mu - H_{OU}(t)) + \sigma\eta_{1/2}(t), \quad (\text{A3})$$

where $\{\eta_{1/2}(t), t \geq 0\}$ is the Gaussian white noise and $\mu \in \mathbb{R}$, $\lambda > 0$ and $\sigma > 0$ are the parameters of the process. On the other hand, the OU process can be obtained from a Brownian motion by the so-called Lamperti transformation.⁷⁹

When $H_{OU}(0)$ is constant and $\mu = 0$, the ACVF of the considered process is given by

$$\mathbb{E}[H_{OU}(t)H_{OU}(s)] = \frac{\sigma^2}{\lambda} e^{-\lambda|t-s|}. \quad (\text{A4})$$

The reflected Ornstein–Uhlenbeck process is defined similarly as the reflected Brownian motion defined in (1). Specifically, at every given time t , when a trajectory exceeds a given threshold, it is reflected.

The parameterization for trajectories' simulation is that $\mu = H_0$ and $H_{OU}(0) = H_0$.

3. Smooth telegraph process

The telegraph process (asymmetric) $\{H_T(t), t \geq 0\}$ is defined as follows:⁸⁰

$$H_T(t) = h_0(-1)^{N(t)} + h_1, \quad (\text{A5})$$

where $\{N(t), t \geq 0\}$ is the homogeneous Poisson process⁸¹ with the intensity $\gamma > 0$ and h_0, h_1 are the positive real numbers such that $h_0 < h_1 < 1 - h_0$. The asymmetric telegraph process is a two-stage process taking the values $H_0 = h_1 - h_0$ and $H_1 = h_1 + h_0$ with probabilities $e^{-\gamma t} \sinh(\gamma t)$ and $e^{-\gamma t} \cosh(\gamma t)$, respectively. The

smoothed (filtered) asymmetric telegraph process $\{H_{ST}(t), t \geq 0\}$ is defined through the following stochastic differential equation in the following form:⁸⁰

$$\frac{dH_{ST}(t)}{dt} = -\beta H_{ST}(t) + \beta H_T(t), \quad (\text{A6})$$

where $\beta > 0$ is a parameter of the filter and $\{H_T(t), t \geq 0\}$ is an asymmetric telegraph process defined in Eq. (A5).

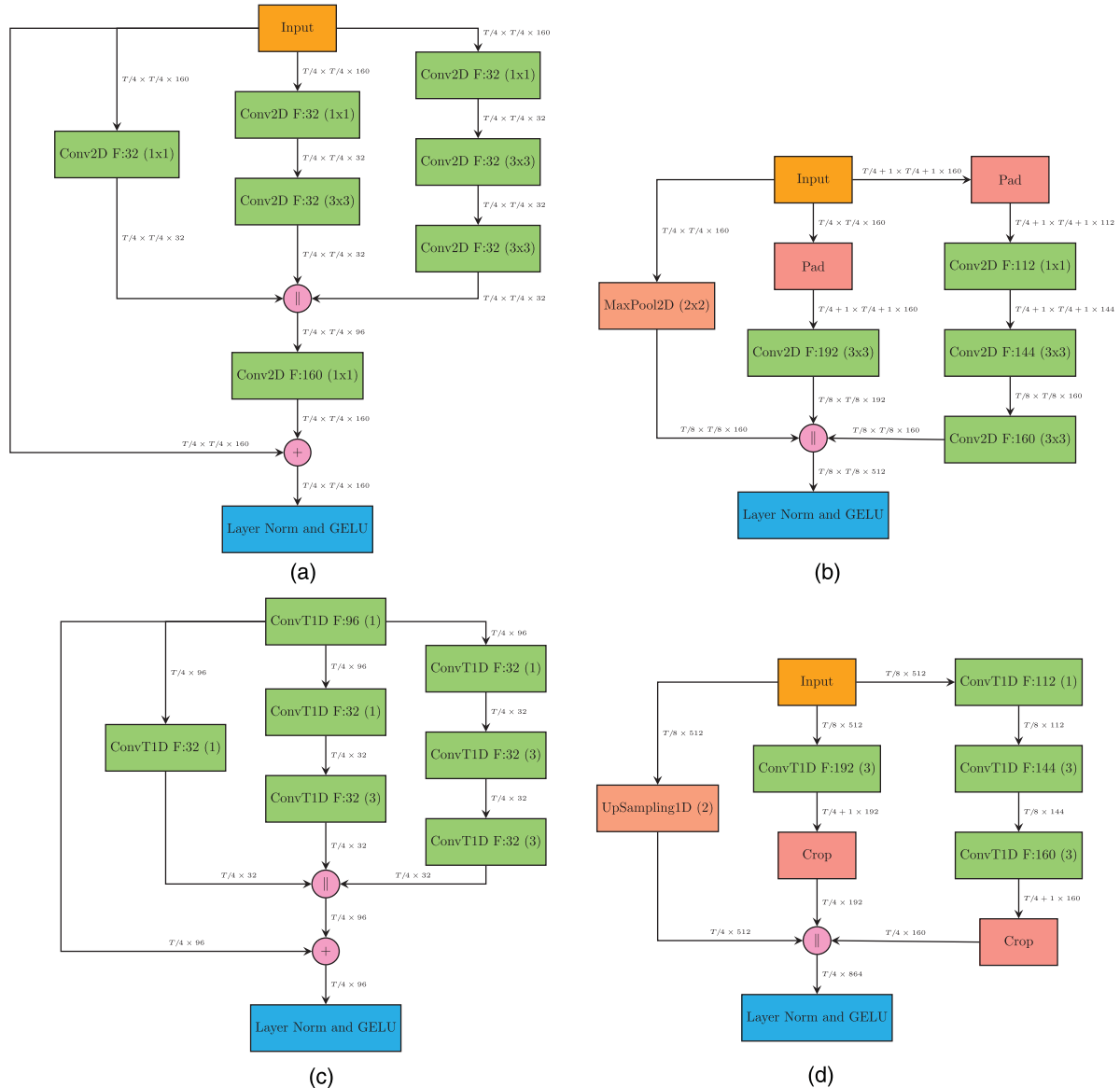


FIG. 10. Blocks of type A. (a) Inception-ResNet-A. (b) Inception-ResNet-Reduction-A. (c) Transpose Inception-ResNet-A. (d) Transpose Inception-ResNet-Reduction-A.

APPENDIX B: DETAILS OF NEURAL NETWORK'S ARCHITECTURE

Building blocks of the architecture presented in Fig. 1 are described in detail in the following figures:

- (a) Fig. 13(a)—Stem,
- (b) Fig. 10(a)—Block A,
- (c) Fig. 10(b)—Reduction Block A,
- (d) Fig. 11(a)—Block B,

- (e) Fig. 11(b)—Reduction Block B,
- (f) Fig. 12(a)—Block C,
- (g) Fig. 12(b)—Transpose Block C,
- (h) Fig. 13(c)—Max-Reduce Block,
- (i) Fig. 11(d)—Transpose Reduction Block B,
- (j) Fig. 11(c)—Transpose Block B,
- (k) Fig. 10(d)—Transpose Reduction Block A,
- (l) Fig. 10(c)—Transpose Block A,
- (m) Fig. 13(b)—Transpose Stem.

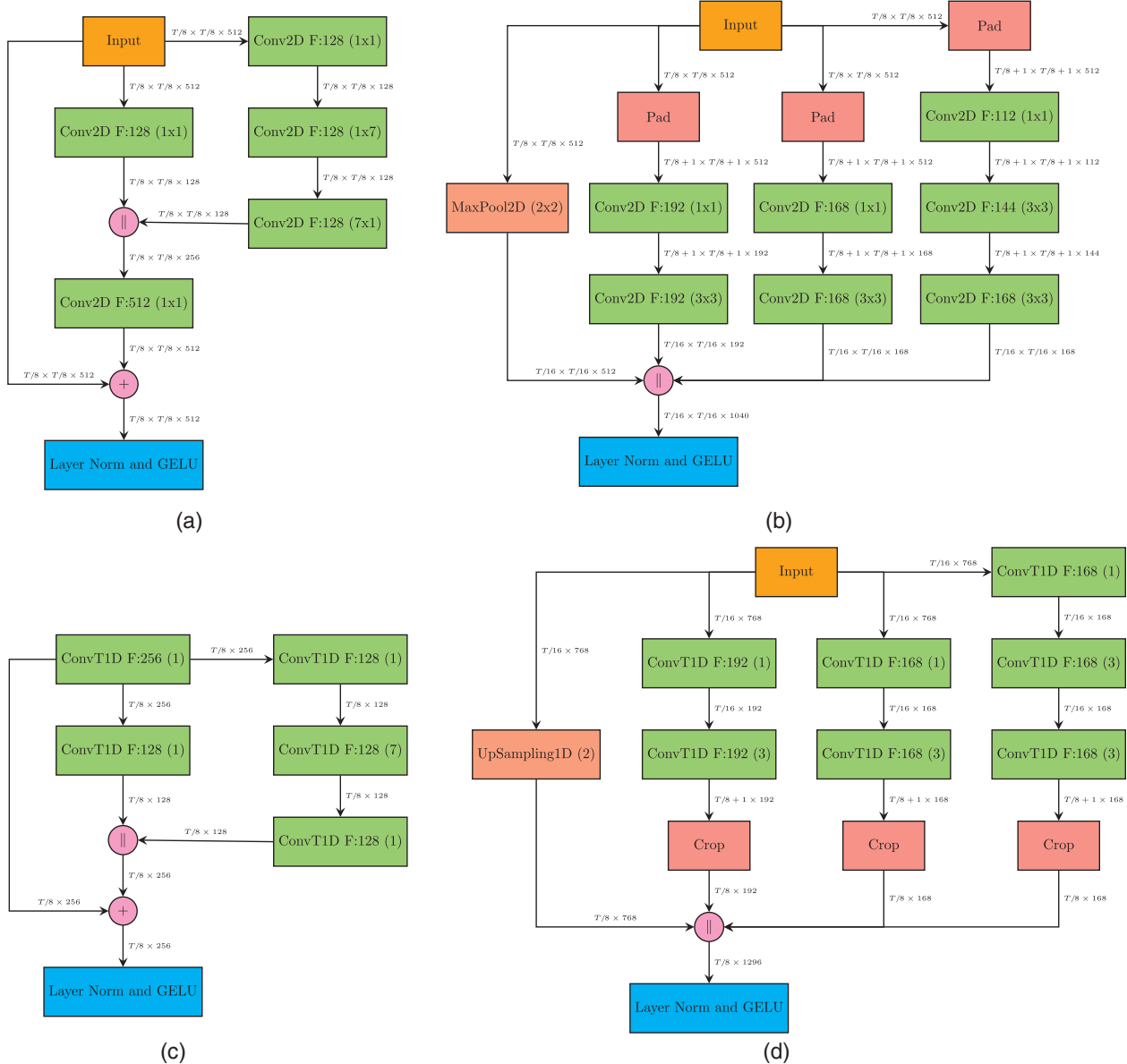


FIG. 11. Blocks of type B. (a) Inception-ResNet-B. (b) Inception-ResNet-Reduction-B. (c) Transpose Inception-ResNet-B. (d) Transpose Inception-ResNet-Reduction-B.

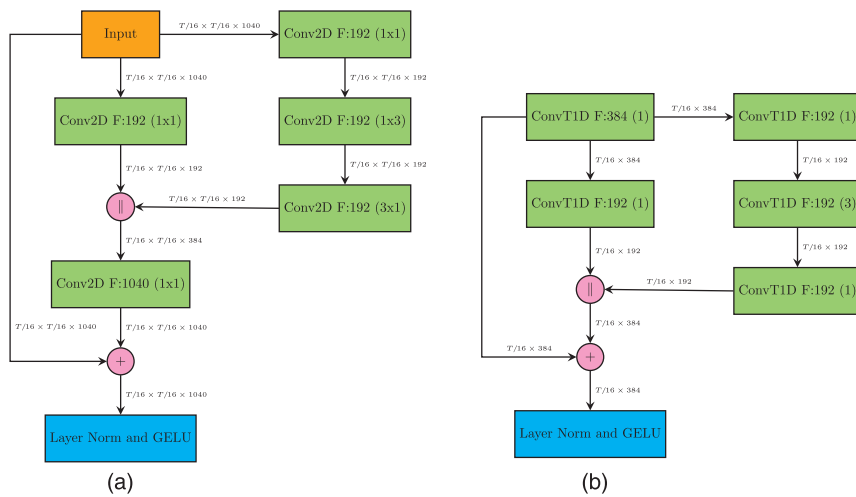


FIG. 12. Blocks of type C. (a) Inception-ResNet-C. (b) Transpose Inception-ResNet-C.

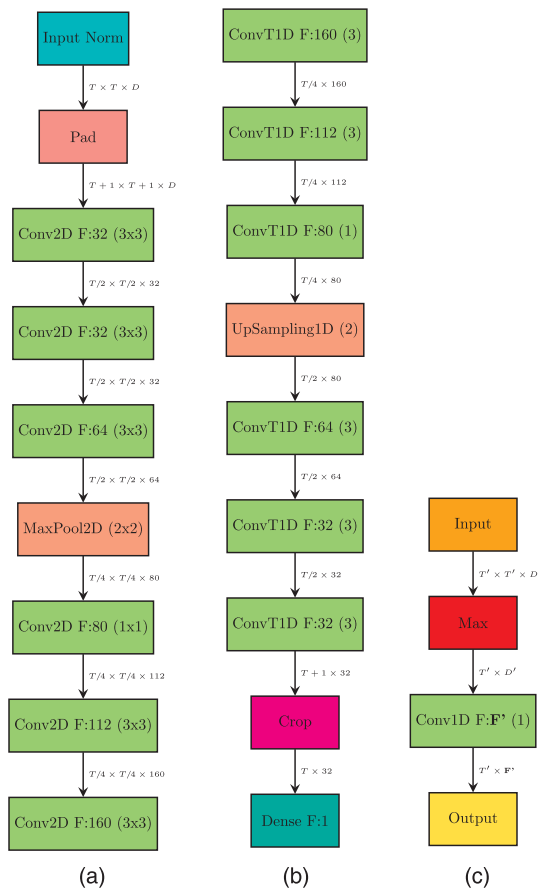


FIG. 13. Stem blocks and Max-Reduce block. (a) Stem. (b) Transpose Stem. (c) Max-Reduce.

REFERENCES

- ¹A.-L. Barabási, "The architecture of complexity," *IEEE Control Syst. Mag.* **27**(4), 33–42 (2007).
- ²N. Molkenhuth, K. Rehfeld, N. Marwan, and J. Kurths, "Networks from flows—From dynamics to topology," *Sci. Rep.* **4**, 4119 (2014).
- ³N. Ekhtiari, A. Agarwal, N. Marwan, and R. V. Donner, "Disentangling the multi-scale effects of sea-surface temperatures on global precipitation: A coupled networks approach," *Chaos* **29**, 063116 (2019).
- ⁴U. Frey, T. Brodbeck, A. Majumdar, D. Robin Taylor, G. Ian Town, M. Silverman, and B. Suki, "Risk of severe asthma episodes predicted from fluctuation analysis of airway function," *Nature* **438**, 667–670 (2005).
- ⁵I. Jabłoński, R. Morello, and J. Mroczka, "The complexity and variability mapping for prediction and explainability of the sleep apnea syndrome," *IEEE Sens. J.* **21**, 14203–14212 (2021).
- ⁶S. Wanqing, X. Chen, C. Cattani, and E. Zio, "Multifractional Brownian motion and quantum-behaved partial swarm optimization for bearing degradation forecasting," *Complexity* **2020**, 8543131.
- ⁷A. V. Weigel, S. Ragi, M. L. Reid, E. K. Chong, M. M. Tamkun, and D. Krapf, "Obstructed diffusion propagator analysis for single-particle tracking," *Phys. Rev. E* **85**, 041924 (2012).
- ⁸T. Westerhold, N. Marwan, A. J. Drury, D. Liebrand, C. Agnini, E. Anagnostou, J. S. K. Barnett, S. M. Bohaty, D. De Vleeschouwer, F. Florindo, T. Frederichs, D. A. Hodell, A. E. Holbourn, D. Kroon, V. Lauretano, K. Littler, L. J. Lourens, M. Lyle, H. Pälike, U. Röhl, J. Tian, R. H. Wilkens, P. A. Wilson, and J. C. Zachos, "An astronomically dated record of Earth's climate and its predictability over the last 66 million years," *Science* **369**, 1383–1387 (2020).
- ⁹N. Dioguardi, F. Grizzi, B. Franceschini, P. Bossi, and C. Russo, "Liver fibrosis and tissue architectural change measurement using fractal-rectified metrics and Hurst's exponent," *World J. Gastroenterol.* **12**, 2187 (2006).
- ¹⁰M. Gilmore, C. Yu, T. Rhodes, and W. Peebles, "Investigation of rescaled range analysis, the Hurst exponent, and long-time correlations in plasma turbulence," *Phys. Plasmas* **9**, 1312–1317 (2002).
- ¹¹X. Liu, B. Wang, and L. Xu, "Statistical analysis of Hurst exponents of essential/nonessential genes in 33 bacterial genomes," *PLoS One* **10**(6), e0129716 (2015).
- ¹²J. Beran, *Statistics for Long-Memory Processes* (Chapman & Hall, 1994).
- ¹³P. Doukhan, G. Oppenheim, and M. Taqqu, *Theory and Applications of Long-Range Dependence* (Birkhäuser, Inc., Boston, MA, 2003).
- ¹⁴A. N. Kolmogorov, "Wiensche spiralen und einige andere interessante Kurven in Hilbertschen Raum, C. R. (Doklady)," *Acad. Sci. URSS (N.S.)* **26**, 115–118 (1940).

of the IEEE Conference on Computer Vision and Pattern Recognition (IEEE, 2015), pp. 1–9.

⁶³K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (IEEE, 2016), pp. 770–778.

⁶⁴I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning* (MIT Press, 2016).

⁶⁵R. Caruana, S. Lawrence, and C. Giles, “Overfitting in neural nets: Backpropagation, conjugate gradient, and early stopping,” *Adv. Neural Inf. Process. Syst.* **13**, 402–408 (2000).

⁶⁶A. Dosovitskiy, P. Fischer, E. Ilg, P. Hausser, C. Hazirbas, V. Golkov, P. Van Der Smagt, D. Cremers, and T. Brox, “FlowNet: Learning optical flow with convolutional networks,” in *Proceedings of the IEEE International Conference on Computer Vision* (IEEE, 2015), pp. 2758–2766.

⁶⁷D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” [arXiv:1412.6980](https://arxiv.org/abs/1412.6980) (2014).

⁶⁸G. Lample, M. Ballesteros, S. Subramanian, K. Kawakami, and C. Dyer, “Neural architectures for named entity recognition,” [arXiv:1603.01360](https://arxiv.org/abs/1603.01360) (2016).

⁶⁹X. Mao, Q. Li, H. Xie, R. Y. Lau, Z. Wang, and S. Paul Smolley, “Least squares generative adversarial networks,” in *Proceedings of the IEEE International Conference on Computer Vision* (IEEE, 2017), pp. 2794–2802.

⁷⁰S. Ruder, “An overview of gradient descent optimization algorithms,” [arXiv:1609.04747](https://arxiv.org/abs/1609.04747) (2016).

⁷¹S. Sabour, N. Frosst, and G. E. Hinton, “Dynamic routing between capsules,” *Adv. Neural Inf. Process. Syst.* **30**, 3856–3866 (2017).

⁷²G. Chan and A. T. Wood, “Simulation of multifractional Brownian motion,” in *COMPSTAT* (Springer, 1998), pp. 233–238.

⁷³T. Dieker, “Simulation of fractional Brownian motion,” Ph.D. thesis, master’s thesis, Department of Mathematical Sciences (University of Twente, 2004).

⁷⁴C. Gosse and V. Croquette, “Magnetic tweezers: Micromanipulation and force measurement at the molecular level,” *Biophys. J.* **82**, 3314–3329 (2002).

⁷⁵D. Szarek, K. Maraj-Zygmunt, G. Sikora, D. Krapf, and A. Wyłomańska, “Statistical test for anomalous diffusion based on empirical anomaly measure for Gaussian processes,” *Comput. Stat. Data Anal.* **168**, 107401 (2022).

⁷⁶F. C. MacKintosh and C. Schmidt, “Microrheology,” *Curr. Opin. Colloid Interface Sci.* **4**, 300–307 (1999).

⁷⁷A. Einstein, *Investigations on the Theory of Brownian Movement* (Dover Publications, New York, 1956).

⁷⁸G. E. Uhlenbeck and L. S. Ornstein, “On the theory of the Brownian motion,” *Phys. Rev.* **36**, 823 (1930).

⁷⁹P. Cheridito, H. Kawaguchi, and M. Maejima, “Fractional Ornstein-Uhlenbeck processes,” *Electron. J. Probab.* **8**, 1–14 (2003).

⁸⁰R. Pawłaj, “Statistical geometry of the smoothed random telegraph signal,” *Int. J. Control* **16**, 629–640 (1972).

⁸¹J. F. C. Kingman, *Poisson Processes* (The Clarendon Press, Oxford University Press, 1993).