scientific reports

Check for updates

OPEN

Generative inpainting of incomplete Euclidean distance matrices of trajectories generated by a fractional Brownian motion

Alexander Lobashev[1], Dmitry Guskov[1] & Kirill Polovnikov[1,2,3]✉

Fractional Brownian motion (fBm) exhibits both randomness and strong scale-free correlations, posing a challenge for generative artificial intelligence to replicate the underlying stochastic process. In this study, we evaluate the performance of diffusion-based inpainting methods on a specific dataset of corrupted images, which represent incomplete Euclidean distance matrices (EDMs) of fBm across various memory exponents (*H*). Our dataset reveals that, in the regime of low missing ratios, data imputation is unique, as the remaining partial graph is rigid, thus providing a reliable ground truth for inpainting. We find that conditional diffusion generation effectively reproduces the inherent correlations of fBm paths across different memory regimes, including sub-diffusion, Brownian motion, and super-diffusion trajectories, making it a robust tool for statistical imputation in cases with high missing ratios. Moreover, while recent studies have suggested that diffusion models memorize samples from the training dataset, our findings indicate that diffusion behaves qualitatively differently from simple database searches, allowing for generalization rather than mere memorization of the training data. As a biological application, we utilize our fBm-trained diffusion model to impute microscopy-derived distance matrices of chromosomal segments (FISH data), which are incomplete due to experimental imperfections. We demonstrate that our inpainting method outperforms standard bioinformatic methods, suggesting a novel physics-informed generative approach for the enrichment of high-throughput biological datasets.

Diffusion probabilistic models are gaining popularity in the field of generative machine learning due to their ability to synthesize diverse and high-quality images from the training distribution. The iterative denoising approach taken by diffusion[1–3] outperforms in quality of generated samples the previously used schemes[4], such as VAEs[5,6] and GANs[7–9], and has demonstrated a distinctive potential in scalability[10]. Recently, several conditional diffusion-based generation methods have been developed[11–13], allowing for effective inpainting of masked images using the pre-trained unconditional diffusion model. Still, whether the diffusion-based inpainting can learn and reproduce the intrinsic non-local dependencies in the pixels of the image drawn from a particular statistical ensemble has remained unaddressed. Furtheremore, recent studies by[14,15] suggest that modern text-to-image generative diffusion models, such as Dalle-2[10], Imagen[16], or StableDiffusion[17], tend to recall samples from their training databases, raising questions about their generalization capabilities and bringing up the copyright infringement concerns during the diffusion training process.

In this paper we consider a dataset of incomplete EDMs and propose to approach the EDM completion problem as the image inpainting via conditioning of the Denoising Diffusion Probabilistic Model (DDPM). Importantly, the possibility of existence of the ground truth of the inpainting in the EDM dataset uniquely allows one to evaluate the quality of the conditional generation at the instance level. At high missing ratio, however, the solution of EDM completion does not exist and one has to rely on the ensemble-level metrics such as Fréchet Inception Distance (FID). Here we ask: can the diffusion model learn the intrinsic correlations between the entries of the matrix when an ensemble of such matrices is given and statistically reproduce them upon the inpainting? In order to explore the modern generative models at this novel angle we consider the pairwise distances between the points of a discrete fractional Brownian process (fBm), the simplest Gaussian generalization of a Brownian motion with strong scale-free correlations. The built-in memory in the fBm process can induce a non-Brownian exponent of the second moment (also known as the mean-squared displacement of

[1]Skolkovo Institute of Science and Technology, Moscow, Russia. [2]University of Potsdam, Institute of Physics and Astronomy, D-14476 Potsdam, Germany. [3]Laboratory of Complex Networks, Center for Neurophysics and Neuromorphic Technologies, Moscow, Russia. ✉email: kipolovnikov@gmail.com

a particle undergoing the anomalous diffusion,[18]), which is translated into strong couplings between the pixels in the distance matrix.

Imputation of missing data has recently got a second wind with the development of high-throughput experimental techniques in chromosome biology. Diffusion models have been recently applied to generate and enhance protein and DNA datasets[19–21]. Hi-C and FISH experiments have provided significant new insights into the fractal (non-Brownian) folding of chromosomes[22,23], despite the data being noisy and incomplete[24,25]. In particular, we and others have recently shown that the spatial organization of human chromosomes without loop-extruding complexes (cohesin motors) statistically resembles the ensemble of fractal trajectories with the fractal dimension $d_f = 3$[22,26–30]. Such an ensemble – leaving aside the biophysical principles of such organization – can be modelled as trajectories of a subdiffusive fBm particle with $H = 1/3$[27]. This suggests an important statistical insight for the downstream data analysis[31,32] (also relevant for the worm connectome datasets[33]).

FISH imaging experiments produce datasets that represent matrices of pairwise distances between chromosomal loci on single cells, which are obtained in multiplex microscopy. Thus each matrix corresponds to internal distances within a given chromosomal segment in a given cell. Occasionally, some data in the matrices is masked due to experimental imperfections (biochemistry of the protocol) posing a real challenge for the methods of the downstream analysis. In particular, inference of features of the 3D organization at the single cell level is notoriously obscured by the sparsity of the dataset at hand[25]; the reliable extraction of features from corresponding single-cell datasets poses a significant challenge for algorithms[31,32,34] and is closely linked to the unique number-theoretic properties of sparse matrices[35]. Here we for the first time propose to use the modern generative AI for the inpainting of missing values and completion of experimentally-derived FISH matrices. For this aim we deploy the pre-trained fBm diffusion benchmark at $H = 1/3$, thus virtually taking into account the intrinsic correlations present in the fractal chromosome trajectories[26,27].

The structure of this paper is as follows. In section Background we formulate the EDM completion problem as the image inpainting task, discuss some of classical results from discrete mathematics related to the existence and uniqueness of the EDM completion. In the following section (a) we demonstrate that the unconditional diffusion generation can learn and reproduce non-local correlations in the images (i.e., matrices) representing EDMs of fBm at various memory exponents *H*: for subdiffusion ($H < 1/2$), normal ($H = 1/2$) and superdiffusion ($H > 1/2$) in the single framework; (b) we apply the diffusion-based inpainting for the EDM completion problem showing that it results in low-rank solutions with the proper fBm-like statistics. We further demonstrate that the diffusion generation is qualitatively different from the database search, regardless of the database size, which is being in contrast with the most recent studies[14,15]. Finally, in the last section, we illustrate how the pre-trained fBm diffusion model can be applied for the imputation of missing values in the chromosomal distance matrices derived from FISH experiments. We demonstrate superior performance of the diffusion-based inpaiting as compared to classical bioinformatics approaches. Our results pave the way for an accurate quantification of the cell-to-cell variability in the genome folding and, broadly, showcase the importance of generative AI in the omics data analysis.

Background

Euclidean distance matrices

In this paper we deal with $n \times n$ matrices A of squares of pairwise distances between n points $x_1, x_2, ..., x_n$ in the D-dimensional Euclidean space. For the purposes of this paper, we considered the case of $D = 3$. Such matrices $A = \{a_{ij}\}$ satisfying

$$a_{ij} = ||x_i - x_j||^2, \quad x_i \in \mathbb{R}^D \tag{1}$$

are called Euclidean distance matrices (EDM). Clearly, A is a symmetric ($a_{ij} = a_{ji}$) and hollow ($a_{ii} = 0$) matrix with non-negative values, $a_{ij} \geq 0$. As a distance matrix in Euclidean space it further satisfies the triangle inequality, $\sqrt{a_{ij}} \leq \sqrt{a_{in}} + \sqrt{a_{nj}}$. The latter constraint imposes essential non-linear relationships between the entries of an EDM, making its rank r independent of n for sufficiently large amount of points n in general position, i.e.

$$r = \min(n, D + 2). \tag{2}$$

Any uncorrupted (complete, noise-less and labelled[36]) EDM A allows for the unique reconstruction of the original coordinates $\{x_i\}$ up to rigid transformations (translations, rotations and reflections). Such reconstructions are called realizations of A. Due to the straightforward relation between an EDM (see Eq. 1) and the corresponding Gram matrix $g_{ij} = x_i^T x_j$,

$$a_{ij} = g_{ii} - 2g_{ij} + g_{jj}, \tag{3}$$

a realization of the distance matrix a_{ij} consists of the origin ($x_1 = 0$) and the principal square root of the $(n-1) \times (n-1)$ matrix $\tilde{g}_{ij}$

$$\tilde{g}_{ij} = \frac{1}{2}(a_{1i} - a_{ij} + a_{1j}), \tag{4}$$

provided that $\tilde{g}_{ij}$ is positive semidefinite with rank D (for points in general position). The latter is known as the Schoenberg criterion[37,38]. Other classical conditions for the existence of a realization of the complete EDM make use of the relations involving the Cayley-Menger determinants[39] and allow to decide whether this realization

exists in the given space dimension D. Note that the general rank property of EDMs outlined above follows simply from Eq. 3: since the rank of Gram matrix g_{ij} is D and the ranks of the other two terms in the equation is 1, the rank of a_{ij} cannot exceed $D+2$. For other interesting properties of EDMs we refer the reader to classical textbooks on the topic[40–42].

Noisy measurements of pairwise distances notably violate the properties of EDMs discussed above. In this case one is interested in the optimal embedding of the points in the space of desired dimension. For that low-rank approximations by means of SVD or EVD of the Gram-like matrix $\tilde{g}_{ij}$ from Eq. 4 are typically implemented in the spirit of the classical multidimensional scaling approach[36,43].

Reconstruction of incomplete EDMs

A case of incomplete distance matrix, where a particular set of pairwise distances in Eq. 1 is unknown, is a prominent setting of EDM corruption that we study in this paper.

EDM completion problem. Let us specify $m < \binom{n}{2}$ missing pairwise distances between n points $x_1, x_2, ..., x_n$ of the D-dimensional Euclidean space by means of the symmetric mask matrix $B = \{b_{ij}\}$ consisting of $2m+n$ zeros and $n^2 - n - 2m$ ones. By definition, $b_{ij} = 0$ if the distance between i and j is unknown or $i = j$, and $b_{ij} = 1$ otherwise. The matrix B is the adjacency matrix of the resulting partial graph. That is, we have a matrix $\tilde{A} = \{\tilde{a}_{ij}\}$:

$$\tilde{a}_{ij} = a_{ij} b_{ij}^{-1} \quad \text{for} \quad b_{ij} = 1, \tag{5}$$

while $\tilde{a}_{ij}$ is undefined where $b_{ij} = 0$. The goal is, given an incomplete matrix $\tilde{A}$, defined by Eq. 5, restore $\binom{n}{2} - m$ missing pairwise distances, while preserving the known m distances. For approximate EDM completions, one seeks an EDM matrix A, such that the following Frobenius norm is minimized:

$$||B \odot (A - \tilde{A})||_F^2 \to \min . \tag{6}$$

Here $\odot$ is the Hadamard product of two matrices, i.e. a $n \times n$ matrix $Z_{ij} = (X \odot Y)_{ij} = X_{ij} Y_{ij}$. The condition Eq. 6 suggests that one is looking for an approximation A of the matrix $\tilde{A}$ at the known entries such that A is an EDM matrix. For precise completions, if they exist, the norm Eq. 6 simply equals to zero.

Clearly, for the EDM completion to exist the matrix $\tilde{A}$ must satisfy all the key properties of EDMs over the known entries, such as symmetricity, hollowness, non-negativity, as well as the triangle inequality at the known triples (i, j, k), s.t. $b_{ij} b_{ik} b_{jk} = 1$. If these trivial conditions are satisfied, the matrix $\tilde{A}$ is called a partial EDM (i.e. every fully specified principal submatrix of $\tilde{A}$ is itself an EDM).

However, this condition is not sufficient. For instance, when graph B contains a cycle with at least four nodes that lacks a chord (an edge connecting two nodes within the cycle but not part of the cycle), one can select distances along the cycle such that the partial Euclidean distance matrix $\tilde{A}$ cannot be completed. Graphs that do not have such cycles are referred to as chordal. The classical GJSW theorem[44] states that a partial EDM $\tilde{A}$ can be completed if the corresponding graph B is chordal, meaning it does not contain holes or cycles of length $l \geq 4$ without chords. Nevertheless, even when the conditions for completion are met, the solution may not be unique.

When is the completion unique?

In this paper we consider a specific case, when the incomplete matrix $\tilde{A}$ is obtained from a full EDM matrix A by masking some (m) distances. Thus, the solution of EDM completion of $\tilde{A}$ surely exists. A key relevant question for us, essential for proper interpretation of the diffusion predictions, is whether the solution of the matrix completion is unique.

In fact, uniqueness of the EDM completion is equivalent to uniqueness of the distance-preserving immersion $\{x_i\}$ of the partial graph B with the lengths of edges $\tilde{A}$ into the metric space $\mathbb{R}^D$. The corresponding bar-and-joint framework $(B, \{x_i\})$ is called rigid (universally rigid), if all these configurations (in any space dimension D) are equivalent up to distance-preserving transformations[45,46]. There has been a series of classical results addressing sufficient conditions for the rigidity of frameworks through the stress and rigidity matrices (see[45] for review), as well as by evoking semi-definiteness of A instead of the non-negativity[47–49]. Still, in practice testing for rigidity is known to be a NP-hard problem, unless the points are in general position[46,50].

The more distances m missing in $\tilde{A}$, the more probable the solution of EDM completion is non-unique. Each ensemble of partial EDMs with exactly $2m$ missing entries can be characterized by the following missing ratio

$$\mu = 2m/(n(n-1)). \tag{7}$$

This number reflects the typical amount of constraints per each vertex (μn) in the partial graph. Though a useful intuitive measure, the missing ratio alone is not telling of the graph rigidity. In particular, in D dimensions *it is not sufficient* to have D constraints per vertex to ensure the graph rigidity (as a counterexample, imagine two cliques of size $> D$ glued at a single vertex).

A greedy algorithm to check for the graph rigidity

Here we suggest the following greedy algorithm that checks for the rigidity of a given graph B. We describe it for $D = 3$, however, it can be simply generalized for an arbitrary D.

The algorithm sequentially chooses and adds vertices one by one to a subgraph, ensuring the growing subgraph at each step remains rigid. The key idea is that the coordinates of a new vertex in D dimensions can be

uniquely determined, if it is connected to at least $D+1=4$ vertices of the rigid subgraph. Following this idea, on the initial step (i) the algorithm identifies the maximal clique with not less than 4 vertices. As the clique has a complete EDM, there is the corresponding unique realization in the metric space (all cliques are rigid). Then, (ii) the algorithm seeks and adds a new external vertex, maximally connected with the rigid subgraph, but having not less than 4 edges. This process continues (iii-iv...) until all vertices are included to the subgraph, or none can be further added. If all the vertices are eventually included, the algorithm confirms that the given graph B is rigid and the EDM completion of $\tilde{A}$ is unique. See Fig. S1 for the graphical sketch and Methods for the pseudo-code of the algorithm.

Fractional Brownian motion

Fractional Brownian motion is one the simplest generalizations of Brownian motion that preserves Gaussianity of the process, but introduces strong memory effects[51]. By definition, fBm is a Gaussian process $B_H(t)$ on the interval [0, T] that starts at the origin, $B(0)=0$, and has the following first two moments:

$$\begin{aligned}&\langle B_H(t)\rangle = 0;\\ &\langle B_H(t)B_H(t')\rangle = \frac{1}{2}\left(t^{2H}+t'^{2H}-|t-t'|^{2H}\right)\end{aligned} \tag{8}$$

and $0<H<1$ is the Hurst parameter (the memory exponent). As it remains Gaussian and ergodic[18,52], the fBm model of anomalous diffusion often allows for analytical treatment. Clearly, the increments of fBm are not independent for $H\neq 1/2$, at which it reduces to Brownian motion. The mean-squared displacement of fBm, $\langle B_H^2(t)\rangle$, describing how far the trajectory spreads from the origin at time t, can be obtained from Eq. 8:

$$\langle B_H^2(t)\rangle = t^{2H}. \tag{9}$$

The parameter H physically characterizes fractality ("roughness") of the trajectory. The autocorrelation of the fBm increments is given by the second derivative of Eq. 9:

$$\langle dB_H(t)dB_H(0)\rangle \sim \frac{2H(2H-1)}{t^{2-2H}}, \quad H=1/2\pm\varepsilon. \tag{10}$$

Thus, normal diffusion (delta-functional correlations) is a particular case of fBm at $H=1/2$, subdiffusion (negative power-law correlations between the increments) corresponds to $H<1/2$, while superdiffusion (positive power-law correlations) corresponds to $H>1/2$.

In this paper we consider a discrete-time process $\{x_i\}_{i=1}^n$ parameterized by $\{s_i\}_{i=1}^n$ along the trajectory with the fBm statistics, Eq. 8 ($x_0=0$ at $s_0=0$). That is, the mean-squared displacement is $\langle r^2(s)\rangle = a^2 s^{2H}$, where a is the typical displacement at a single "jump"; the pdf of the end-to-end vector $r=x_i-x_j$ between the points s_i and s_j of the trajectory depends only on the contour distance $s=|s_i-s_j|$ and reads

$$P(r|s) = \left(\frac{D}{2\pi a^2 s^{2H}}\right)^{D/2} \exp\left(-\frac{Dr^2}{2a^2 s^{2H}}\right). \tag{11}$$

Anomalous diffusion featuring strong memory effect can have different physical origins; examples include charge transport in semiconductors, cellular and nuclear motion etc. (see[52,53] for a comprehensive review). Recently, an fBm model of chromosome folding has been proposed, where fractal chromosome conformations are modelled as trajectories of a subdiffusive fBm particle[26–28].

In what follows, we train the diffusion model to generate complete EDM images $A=\{a_{ij}\}_{i,j=1}^n$ (see Eq. 1) of discrete fBm trajectories $\{x_i\}_{i=1}^n$ for different values of the Hurst exponent. The essential new competence of the diffusion generative model that we test is its ability to learn the power-law correlations in the fBm trajectories. This is crucial for the proper inpainting of the missing data in the distance matrix.

Diffusion-based generation of fBm trajectories

Diffusion generative models are renowned for their impressive capabilities to generate diverse natural images from a specified distribution. In essence, a Denoising Diffusion Probabilistic Model (DDPM, see Methods) produces data by reversing a forward noising process[1,2]; it takes a matrix of pure Gaussian noise (composed of independent normally distributed entries) as input and outputs a matrix that conforms to the target distribution (e.g., resembling a portrait of a human). Generation occurs through a series of sampling steps guided by the optimal gradient learned during the forward noising stage, facilitating a rich diversity of outputs. Thus, reversing the noise along the learned "pathway" enables the creation of a wide array of new samples from the original distribution. In the following subsection, we extend these concepts of (unconditional) generation to create a distinctive ensemble of Euclidean distance matrices (EDMs) with specific statistical characteristics. Subsequently, we apply the trained unconditional DDPM to address the problem of inpainting incomplete EDMs.

Unconditional generation

We aim to train DDPM to generate realistic ensembles of distance matrices with proper fBm statistics, see Figure 1(a)-(b). For the training set we synthesize $M=2\times 10^5$ random fBm trajectories $\{x_i\}_{i=1}^n$ of length $n=64$ in $D=3$ using Davies-Harte algorithm[54] for three values of the Hurst parameter: $H=\frac{1}{3}$(subdiffusion), $H=\frac{1}{2}$(normal diffusion) and $H=\frac{2}{3}$(superdiffusion). Then we train the DDPM on the corresponding dataset

of complete EDMs $A = \{a_{ij}\}_{i,j=1}^{n}$(see Eq. 1), obtained from the synthesized fBm trajectories (i.e., a separate round of training for each Hurst parameter). During each training round each of M matrices from the training ensemble is gradually transformed to white noise, while the mean reverse gradient is learned through training of a neural network. Finally, one initiates the denoising procedure and uses the learned gradients to transform a matrix of pure noise into a matrix from the original ensemble of Euclidean distance matrices (EDMs), preserving the statistical characteristics of the corresponding fractional Brownian motion (fBm).

Figure 2(a) demonstrates the generated distance matrices for the three representative values of H with the respective snapshots of trajectories. Qualitatively, with decrease of H trajectories turn to be more compact, enriching the distance matrices with local patterns. To ensure that the generated ensemble of Euclidean distance matrices (EDMs) accurately reflects the proper fractional Brownian motion (fBm) statistics, we compute the ensemble-averaged scaling of the mean end-to-end squared distance $\langle r_H^2(s)\rangle^{1/2}$ for segments of length s by averaging along the diagonal s in matrices from the generated ensemble. The log-log plot in Figure 2(b) illustrates that the DDPM model successfully reproduces the power-law scaling $\sim s^H$ of the distances for all s. Additionally, it accurately learns the correct memory exponent H in all the regimes within the statistical error. Some subtle deviations at small H can be attributed to insufficient training. Consequently, the generated EDM matrices exhibit trajectories with fractal properties analogous to those of the training ensemble. Next, we examine whether not only the scaling but also the entire distribution of pairwise distances is preserved in the generated matrices, see Figure 2(c). We find that the probability densities of the rescaled pairwise distance $|r|/\langle r_H^2(s)\rangle^{1/2}$ collapse onto the Maxwell form $4\pi r^2 \times P(r|s)$, where $P(r|s)$ is the standard Gaussian distribution. This behaviour occurs independently of the Hurst parameter and segment sizes s, in full agreement with Eq.11.

To assess whether the generated matrices closely approximate the target ensemble of Euclidean distance matrices (EDMs), we compute the similarity between a training sub-ensemble of matrices of size $M/2$ and the generated ensemble of the same size. We then compare this similarity to that between two training sub-ensembles of size $M/2$, as presented in Table 1. For the similarity measure, we utilize the Fréchet Inception Distance (FID), which is widely employed to evaluate the quality of images synthesized by generative models. Essentially, FID calculates the distance between the two distributions of high-level features extracted from the images processed through a convolutional neural network (Inception v3 architecture). A smaller FID value indicates that the two sets of images exhibit a similar diversity of features, signifying greater similarity between them. Table 1 demonstrates that the FID score between the generated and the ground truth ensembles is higher than that between two ground truth sub-ensembles. Nonetheless, the score remains relatively low, with the corresponding noise level not exceeding 1.7%. This indicates that the diffusion process generates a statistically representative ensemble of Euclidean distance matrices (EDMs) with the specified Hurst parameter.

Additionally, we examine whether the matrix rank is accurately reproduced. As discussed in the first section, a set of points in general position in a space of dimension D would yield an EDM matrix of rank $r = D + 2$. The

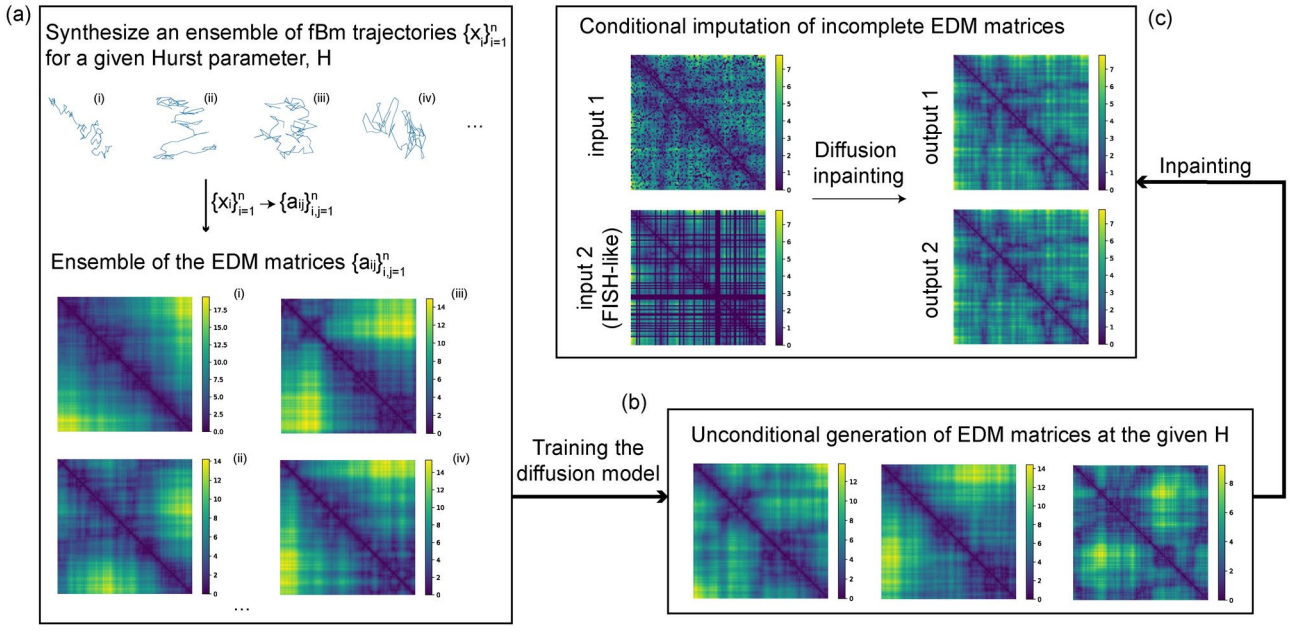

Fig. 1. A pipeline of the diffusion-based inpainting scheme. (**a**) Synthesise of an ensemble of fBm trajectories $\{x_i\}_{i=1}^{n}$ of a certain length n and Hurst parameter H. Each trajectory is then converted into an EDM matrix $\{a_{ij}\}_{i,j=1}^{n}$. (**b**) The Denoising Diffusion Probabilistic Model (unconditional DDPM, Algorithm 1 in Methods) is trained on the ensemble of EDM matrices. (**c**) The trained diffusion model is further used for the inpainting of masked values in incomplete EDMs (conditional generation, Algorithms 2–5 in Methods). Two examples of the input are shown: the masked entries are randomly dispersed in the EDM matrix (top) and entire rows and columns are masked (bottom). The latter example mimics the experimental FISH data.

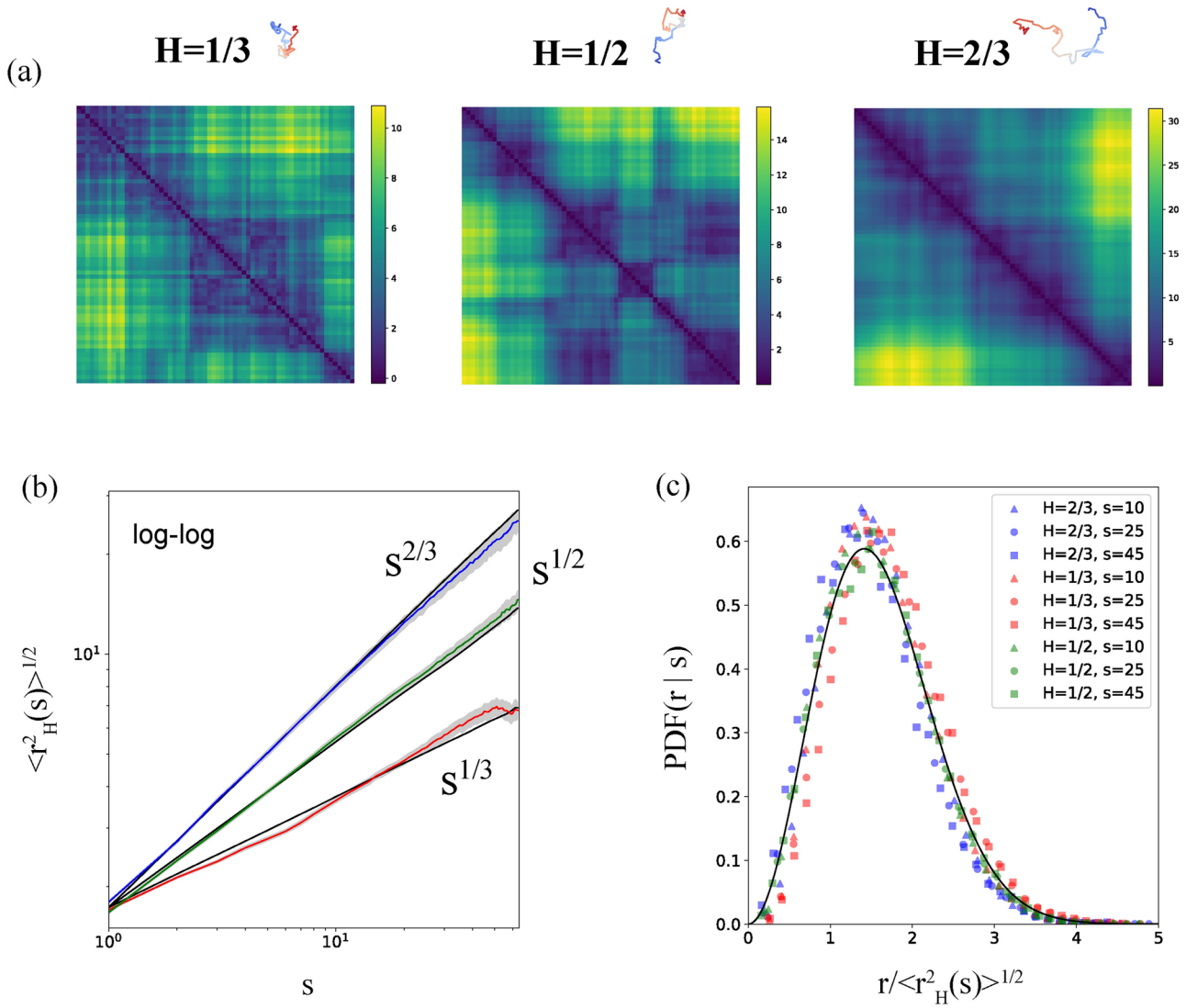

Fig. 2. (**a**) Euclidean distance matrices and the corresponding fBm trajectories, generated by the unconditional diffusion model, for the three values of Hurst exponent: $H = 1/3$(subdiffusion), $H = 1/2$(normal diffusion) and $H = 2/3$(superdiffusion). The trajectories were obtained using gradient optimization of three-dimensional coordinates to match the generated distance matrices. The color changes from red to blue along the trajectory. (**b**) Scaling of the typical distances as a function of the contour length s for $H = 1/3$(red), $H = 1/2$(green) and $H = 2/3$(blue). EDM samples from the generated ensemble fill the grey intervals and thick color lines correspond to the ensemble-averaged curves. Black lines correspond to the training databases. (**c**) Collapsed probability densities of the diffusion-generated pairwise distances between two points on the trajectory separated by contour distance s. The black curve corresponds to the Maxwell distribution $4\pi r^2 \times P(r|s)$, where $P(r|s)$ is the standard Gaussian distribution.

last column in Table 1, which displays the fraction of the first 5 singular values in the nuclear norm of the matrix, clearly indicates that the rank of the generated matrices is close to $r = 5$, as is expected for EDMs in $D = 3$.

All together, the unconditional generation of distance matrices by the pre-trained DDPM reproduces the statistical properties of the ensemble of EDM images corresponding to fBm trajectories, confirming that diffusion generative models are able to learn strong algebraic correlations in the training data (see Eq. 10).

Inpainting of incomplete EDMs

As the diffusion model fairly captures and reproduces the intrinsic correlations in fBm trajectories, we next ask whether the conditional generation (inpainting) of the pre-trained DDPM is able to optimally fill missing data in incomplete EDMs with memory. For the inpainting problem, we generate symmetric binary mask B with a given sparsity μ by sampling a Bernoulli random variable (Figure 1(c), input 1). The probability of getting a 0 is equal to the missing ratio μ for each element in the upper triangle of A. Then, the partially known distance matrices $\tilde{A}$ are generated by multiplying the fully-known distance matrix A by the corruption mask, $\tilde{A} = A \odot B$(compare

	Metric				
Hurst	FID DDPM	FID reference	FID ref. noised	Equiv.noise, %	$\frac{\sqrt{\sum_{i=1}^{5}\lambda_i^2}}{\sqrt{\sum_{i=1}^{64}\lambda_i^2}}$
H = 1/3	0.097 ± 0.012	0.033 ± 0.010	0.098 ± 0.013	1.68 ± 0.15	0.998 ± 0.001
H = 1/2	0.087 ± 0.013	0.037 ± 0.010	0.083 ± 0.011	0.356 ± 0.003	0.9997 ± 0.0003
H = 2/3	0.088 ± 0.023	0.044 ± 0.011	0.094 ± 0.015	0.102 ± 0.001	0.9997 ± 0.0003

Table 1. DDPM unconditional generation. The FID is calculated between an ensemble of synthesized fBm distance matrices (the "ground truth", the size $M/2$) and an ensemble of unconditional DDPM samples of the same size corresponding to three different values of the Hurst parameter. The dimension of the InceptionV3 embedding used is 64. For the reference, we compute the FID between two independently synthesized ensembles of the same number of fBm distance matrices. The noised FID is then calculated between a ground truth fBm ensemble and a fBm ensemble with the multiplicative noise. The level of the multiplicative noise is determined to match the FID values of the DDPM model. The last column reflects the contribution of the first $r = 5$ singular values of the DDPM-generated matrices A in the nuclear norm.

Figure 3(a) and (b)). Consequently, the inpainting task involves reconstructing A given the partially known matrix $\tilde{A}$ and the mask B.

In this work, we test several diffusion-based methods such as DDPM, DDNM[12], DDRM[13], and RePaint[11]. These methods only need a pre-trained unconditional DDPM model as the generative prior, but we stress that DDNM, DDRM, and RePaint additionally require knowing the corruption operators at both training and generation. In our case, the corruption operator is a known corruption mask, which helps diffusion to inpaint. Methods DDNM and RePaint use a time-travel trick (also known as resampling in[11]) for better restoration quality, aimed at intense inpainting with a huge mask, but they can be adversarial at small missing ratio μ. It was shown in[12] that DDNM generalizes DDRM and RePaint, but in our paper, we follow the convention that DDNM is a model with parameters, where the travel length and the repeat times are both set to 3, and for RePaint, we use a number of resamplings set to 10. We further discuss the differences between these methods in the Methods.

Alongside the (i) diffusion based methods DDPM, DDNM, DDRM, RePaint for inpainting[1], we test other approaches. (ii) FISTA is the classical low-rank completion method that exactly recovers unknown distances in the case when such recovery is unique. (iii) Trajectory optimization relies on the direct gradient optimization of 3D coordinates, incorporating a prior that the reconstruction A is an EDM with rank $r = D + 2 = 5$ in general configuration. (iv) The nearest neighbor (NN) method naively fills unknown matrix elements with the closest known distances in the matrix, assuming that close elements are similar. Finally, we also test the (v) database search approach, which fills unknown elements using the entries of the most similar EDM in a pre-generated database (see Methods for the details).

Visual inspection of the inpainted matrices shows remarkable results of the diffusion-based inpainting over classical EDM completion approaches at high sparsity μ(Figure 3, Figure S5). For systematic comparison of different methods, we measure RMSE and FID scores between exact distance matrices and reconstructed ones for 100 values of sparsity, equally spaced from $\mu = 0.01$ to $\mu = 0.99$, see Figure 4. To check for the uniqueness of the EDM completion, for each missing ratio μ we plot a fraction of uniquely recoverable EDMs that pass the rigidity test (see the previous section). We find that in the regime of small μ, where the solution is unique, RMSE of the reconstruction by FISTA is the smallest, and it converges to RMSE of diffusion-based inpainting when the uniqueness is effectively lost at $\mu \approx 0.6 - 0.8$. However, low FID values of the inpainting similar to the ones of FISTA indicate that the diffusion approach yields correct distributions of the matrices, while the intrinsic stochasticity of DDPM is the reason of the larger RMSE. We further find that the inpainting has a significantly smaller FID than the trajectory optimization (OPT) and outperforms in RMSE at small μ. Despite that OPT has the exact rank by construction, it fails to correctly reproduce the fBm-like statistics of the ensemble at small s (see Figure S2). Similar behaviour of the metrics is observed for $H = 1/3$ and $H = 2/3$(Figures S3,S4). Interestingly, among different diffusion-based inpainting schemes, DDRM has the smallest RMSE in a wide range of missing ratio μ, as can be seen in Figure S6 and Table 2. At some point DDRM performance decreases, while DDPM and DDNM generally better behave at very high missing ratios. Our numerical experiments thus demonstrate that DDRM method has the highest precision on the fBm benchmark in the wide range of sparsity.

Additionally, for the diffusion-inpainted matrices with Hurst exponents $H = 1/2$ and $H = 1/3$ we evaluated the physical characteristics of predicted versus ground-truth point clouds. First, we computed the gyration radius Rg^2 of the cloud - a measure of the average pairwise distances between the points - for both the known/masked and inferred pairwise distances. This analysis was conducted at a sufficiently high sparsity level ($\mu = 0.9$), ensuring that the unique solution for inpainting does not exist (Figure S7(a,c)). Despite the high fraction of unknown elements, our results show that diffusion inpainting accurately reproduces the physical size of the object within statistical error. Second, considering the "polymer nature" of the cloud, we analyzed the scaling relationship between physical distance and contour distance along the chain (Figure S7(b,d)). We found that the inferred scaling, derived from the predicted distances, closely aligns with the ground-truth scaling, further confirming the physical consistency of the inpainted matrices. These findings validate the robustness of the inpainting method in preserving key physical properties.

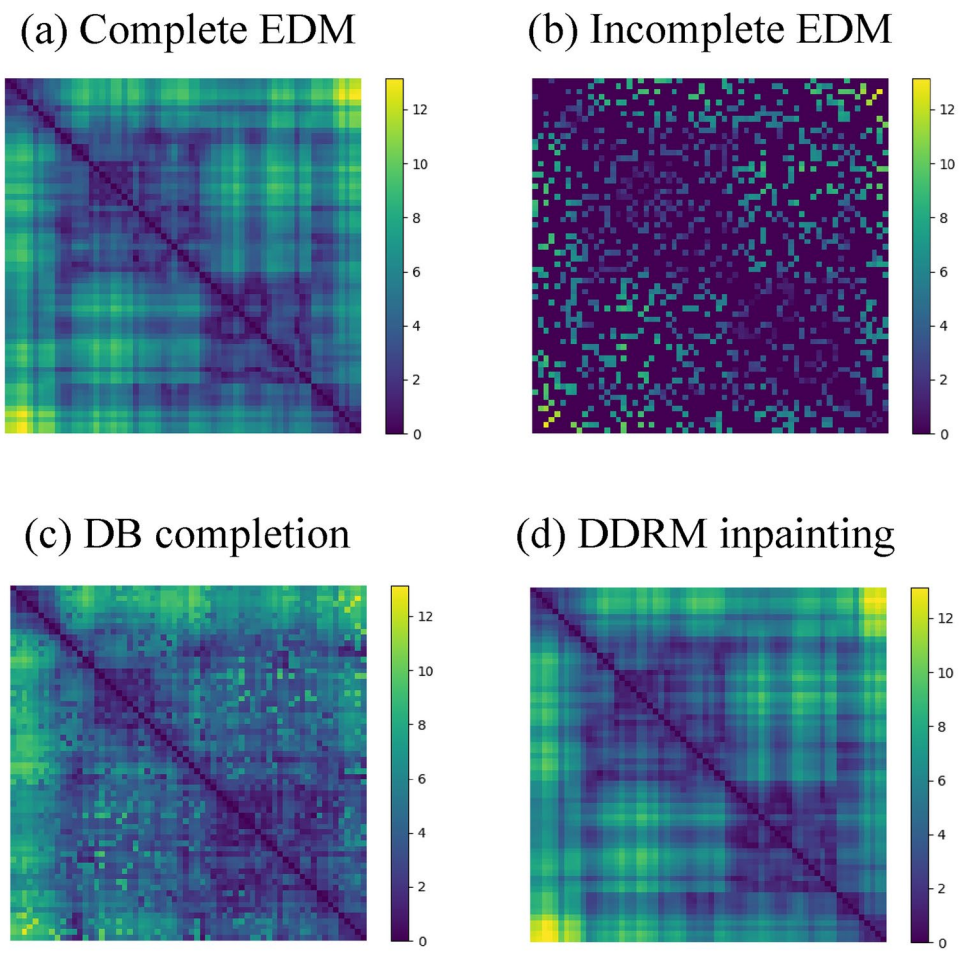

Fig. 3. Examples of EDM completion. (**a**) Complete EDM for a Brownian trajectory ($H = 1/2$). (**b**) A corrupted EDM obtained from (**a**) by masking distances at sparsity $\mu = 0.75$. No exact solution exists at such a high sparsity. (**c**) The most similar matrix from the database search (the database size is $M = 2 \times 10^4$). (**d**) The EDM completion obtained by the DDRM inpainting.

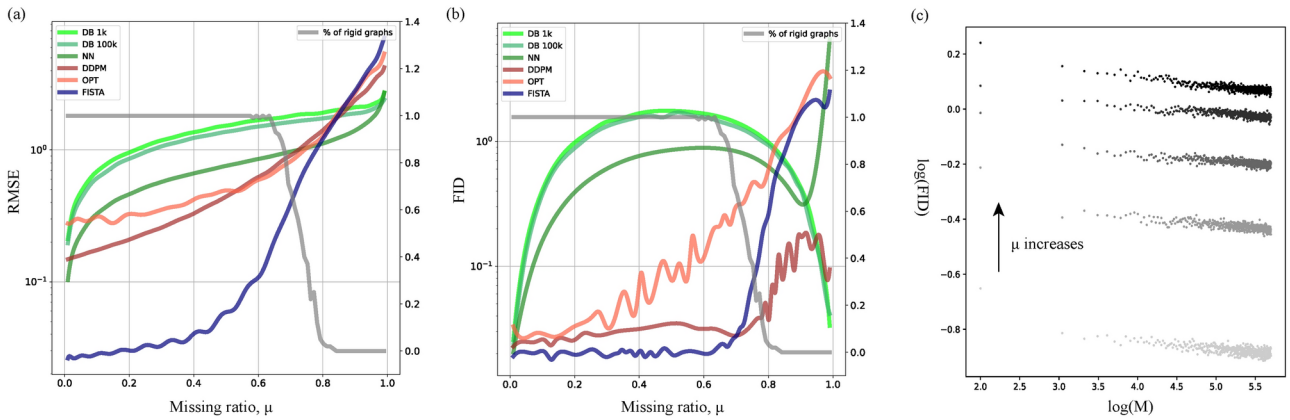

Fig. 4. RMSE (**a**) and FID (**b**) plots as a function of missing ratio μ for different data imputation methods ($H = 1/2$). The fraction of rigid graphs is shown in the second axis (grey). (**c**) Log of FID of the matrices reconstructed via the database search as a function of the log of the database size M. Scatters for different values of sparsity $\mu = 0.05, 0.1, 0.15, 0.19, 0.24$ are shown, corresponding to the regime where the unique EDM completion exists. The number of matrices used in the computation of the FID metrics on (b) is equal to $2 * 10^4$.

Sparsity	Metric	Method				
		RePaint	DDRM	DDNM	DDPM	Database search
$\mu = 0.25$	RMSE↓	0.49 ± 0.02	**0.170 ± 0.017**	0.211 ± 0.018	0.313 ± 0.023	1.12 ± 0.12
	FID↓	0.0446 ± 0.0026	**0.013 ± 0.0017**	0.027 ± 0.0015	0.0235 ± 0.0019	1.225 ± 0.009
	Rank↑	0.858 ± 0.025	0.853 ± 0.023	0.854 ± 0.022	0.849 ± 0.025	0.65 ± 0.05
$\mu = 0.5$	RMSE↓	0.54 ± 0.04	**0.241 ± 0.027**	0.325 ± 0.027	0.55 ± 0.05	1.61 ± 0.18
	FID↓	0.053 ± 0.003	**0.018 ± 0.002**	0.053 ± 0.002	0.0246 ± 0.0007	1.79 ± 0.01
	Rank↑	0.86 ± 0.025	0.854 ± 0.025	0.853 ± 0.025	0.843 ± 0.030	0.63 ± 0.05
$\mu = 0.75$	RMSE↓	0.68 ± 0.06	**0.42 ± 0.04**	0.56 ± 0.04	1.23 ± 0.18	1.97 ± 0.22
	FID↓	0.075 ± 0.003	**0.034 ± 0.003**	0.116 ± 0.002	0.096 ± 0.003	1.25 ± 0.011
	Rank↑	0.863 ± 0.025	0.854 ± 0.027	0.854 ± 0.027	0.82 ± 0.04	0.65 ± 0.05

Table 2. Comparison of different inpainting methods in EDM completion. The FID is calculated between an ensemble of distance matrices, which are generated by the Davies-Harte algorithm, and reconstructed samples corresponding to three different sparsity values μ. The dimension of the InceptionV3 embedding used for FID is 64. The rank measures the contribution of the first $r = 5$ singular values of the reconstructed matrix in the nuclear norm. The number of matrices used in the computation of the FID metrics is equal to 10^5.

Methods	DB 500	DB 1k	DB 2k	DB 5k	DB 20k	DB 200k
FID(train, gen)	0.90 ± 0.24	0.56 ± 0.04	0.341 ± 0.006	0.257 ± 0.008	0.248 ± 0.008	0.0875 ± 0.0005
FID(test, gen)	0.91 ± 0.23	0.53 ± 0.03	0.335 ± 0.008	0.257 ± 0.008	0.251 ± 0.002	0.0874 ± 0.0007

Table 3. FID(train, generated) and FID(test, generated) for uncodnitional DDPM samples for diffusion trained on databases of different sizes.

The main experiments were conducted on matrices of size $N = 64$, which matches the matrix size used during training. Inpainting on larger matrices follows a principle similar to natural image restoration, where diffusion-based models are first trained on smaller resolutions and later fine-tuned for larger ones. However, increasing the matrix size significantly raises GPU memory requirements, limiting the maximum feasible size. In Figure S8(a), we demonstrate the application of our model, trained on 64×64 matrices (Hurst parameter $H = 1/2$), to a larger 256×256 matrix with $\mu = 0.9$ sparsity. To evaluate the model's generalization capability, we further applied it to even larger matrices, as shown in Figure S8(c) for $\mu = 0.25$ sparsity. Our results indicate that the method performs well up to 1024×1024, though reconstruction quality depends on the sparsity level. Up to approximately $\mu = 0.9$, our approach consistently outperforms nearest-neighbor interpolation (Figure S8(b)). Due to GPU memory constraints (30 GB), we limited our study to 1024×1024 matrices. For matrices exceeding GPU memory limits, a practical solution is to divide the large matrix into smaller, overlapping patches. These patches can be processed individually and reassembled to reconstruct the full matrix.

Diffusion vs Database search

A recent study has reported that diffusion can memorize and generate examples from the training dataset[14,15]. However, already for the unconditional generation and for various training database sizes we observe that the FID score between the matrices generated by diffusion and the train dataset is similar to the FID between the generated matrices and an independently synthesized set of distance matrices (Table 3). This suggests that DDPM is able to generalize rather than memorize the training examples.

Next, for the problem of conditional generation we find that different schemes of the diffusion inpainting significantly outperform the database search in all sparsity regimes, see Table 2 and Figure 4. Obviously, in contrast to the inpainting completion that has low rank, the matrix completed with the database would have essentially higher rank (lower fraction of the first 5 singular values, as shown in Table 2). Furthermore, Figure 4(a)-(b) importantly shows that for various database sizes M, the diffusion-based inpainting displays *a different convexity* of RMSE plots and different behaviour of FID. These results clearly suggest that generation by diffusion *qualitatively* differs from the database search, even for the largest database size $M = 10^5$, when Mn^2 approaches the number of parameters of the diffusion model ($\approx 10^8$).

We find that the FID of the database search generation weakly depends on the database size M. The respective slope of FID with M in the double-log scales (FID $\sim M^{-\gamma}$) seems to be independent of the missing ratio μ in the regime where the partial graph remains rigid, see Figure 4(c). By collapsing the plots for different μ together we find the power-law dependence of FID on μ, i.e. FID $\sim \mu^a$ with $a \approx 1.41 \pm 0.06$. For the set of collapsed data at different μ we further estimate the optimal power-law exponent $\gamma \approx 0.026 \pm 0.003$(Figure S9).

$$\mathrm{FID}_{\mathrm{database}}(M, \mu) \sim \mu^a M^{-\gamma};\ a \approx 1.41,\ \gamma \approx 0.026. \tag{12}$$

Then we extrapolate the scaled FID to larger database sizes M to find the effective M^* needed to reproduce the correct distribution of EDMs. Using the FID of the diffusion inpainting as a reference, we numerically obtain a range of magnitudes corresponding to the effective database size

$$\log(M^*)|_{\text{exp}} = 76.7 \pm 10.3. \tag{13}$$

Such a giant size of the effective database M^* as compared to the amount of parameters in the diffusion model is explained by the exponential dependence of M^* on the trajectory length n. Indeed, for a single trial, the expected negative log-likelihood of a random variable x from a standard normal distribution is $E[-\ln(p(x))] = \ln(\sqrt{2\pi}) + \frac{1}{2}$. For the purpose of counting we assume the trajectories are drawn on the lattice with the fixed step size and fixed origin, then, the whole ensemble of distance matrices can be generated with $\sim 2(n-1)$ Gaussian random variables. Therefore, the total entropy of the ensemble, or the log of the effective database size, reads

$$\log(M^*)|_{\text{theory}} = \frac{2(n-1)}{\ln(10)}\left(\ln(\sqrt{2\pi}) + \frac{1}{2}\right) \approx 78.89. \tag{14}$$

As a caveat, note that this theoretical result provides the upper bound for M^*, and it should be reduced upon taking into account the distance-preserving transformations. Nevertheless, comparing Eq. 13 with Eq. 14 we find a remarkable agreement.

Filling missing data in the FISH dataset using the fBm benchmark

As an application of the pre-trained fBm diffusion model, we briefly discuss the results of imputation of missing data in single cell matrices of pairwise distances between chromosomal segments (see Figure 5) obtained from microscopy experiments (Fluorescence In Situ Hybridization, FISH[23]). The corresponding public folder with the raw data is available at the Github page of the paper (https://github.com/BogdanBintu/ChromatinImaging). The dataset represents the 3D coordinates of 30kb segments on a human chromosome 21 (the human colon cancer cell line, HCT116) measured using the multiplex microscopy. Noticeably, some of the coordinates are missed (nan values in the data). For our purposes of the restoration of missed coordinates we used the data with auxin that supposedly corresponds to the condition with no cohesin-mediated loops. Indeed, in vivo data shows that without loops chromosomes exhibit *fractal* statistics, thus justifying the use of fBm trajectories in the training phase of our benchmark[26,27]. We take the 2Mb-long segment from 28Mb to 30Mb for the analysis, the corresponding file name is "HCT116_chr21-28-30Mb_6 h auxin.txt".

First, using the raw data we reproduce the fractal scaling of chromosomal folding[22,26,27], i.e. $\langle r^2(s)\rangle^{1/2} \sim s^{1/3}$, see Figure S10. We stress that such a behaviour corresponding to the fractal dimension $d_f = 3$ is a result of depleted cohesin from the cell nucleus, which is known to extrude loops on chromosomes (see our previous works[26,55,56] for the biophysical model showing how cohesin loops break the scale-invariance). Thus, to inpaint the missing data we decided to use the diffusion model trained on the fBm ensemble with $H = 1/3$(the fractal dimension is the inverse of the Hurst parameter). Figure 5 shows the resulting matrices obtained using various inpainting methods run on a particular FISH dataset (cell 343, see Tables S2-S3 for the data).

To measure the performance of the DDPM inpainting in comparison to other methods we chose 670 cells (out of 7380 cells) in the dataset that have exactly 15 missing rows and columns. This corresponds to sparsity $\mu' = 0.29$. In order to compute RMSE (Table 4) we additionally dropped 10 rows and columns from the matrices resulting in $\mu = 0.63$. We then imputed the missing distances using the standard bioinformatics approaches (nearest neighbor, ensemble mean) and various diffusion-based inpainting methods (DDRM, DDNM, DDPM, RePaint). The nearest neighbor approach relies on filling the unknown distances with the nearest neighbour in the same matrix (cell). The ensemble mean approach fills the missing entry with the corresponding average over the cells where this element is known. The average RMSE was computed for each imputed cell over the known values in the dropped 10 columns and rows. Note that since the *entire* rows and columns are missing in FISH distance matrices, precise EDM completion algorithms such as FISTA or trajectory optimization (OPT) are not applicable here.

Consistently with numerical experiments, the Table 4 demonstrates that the DDRM inpainting trained on the fBm benchmark is superior over other diffusion-based and bioinformatics approaches. It should be noted that it has a comparable RMSE and rank with DDNM inpainting, while RePaint and DDPM behave slightly worse (the resulting RMSE is more than 10% larger). This is to be compared with other approaches, such as filling the missing distances using the nearest neighbor pixel from the same matrix (NN) or using the average over the cells where this matrix element is known (Ens. mean). These clearly naive approaches behave significantly worse both in the rank and RMSE. This observation highlights that the fBm benchmark for diffusion-based inpainting shows evidently better performance on a biological dataset than canonical bioinformatics approaches.

Conclusion

By treating the Euclidean distance matrices as images, in this paper we demonstrated that diffusion probabilistic model can learn the essential large-scale correlations in the distance matrices of the ensemble of fBm trajectories for various memory exponents H. Based on this observation, we apply the diffusion-based inpainting trained on the fBm benchmark for the problem of EDM completion, exploring diffusion inpainting methods at various sparsity parameters. Using our benchmark we observe that the diffusion behaves drastically different from the database search with the database size similar to the number of parameters of the diffusion model. We provide a theoretical argument for the effective database size explaining such a qualitative difference and verify

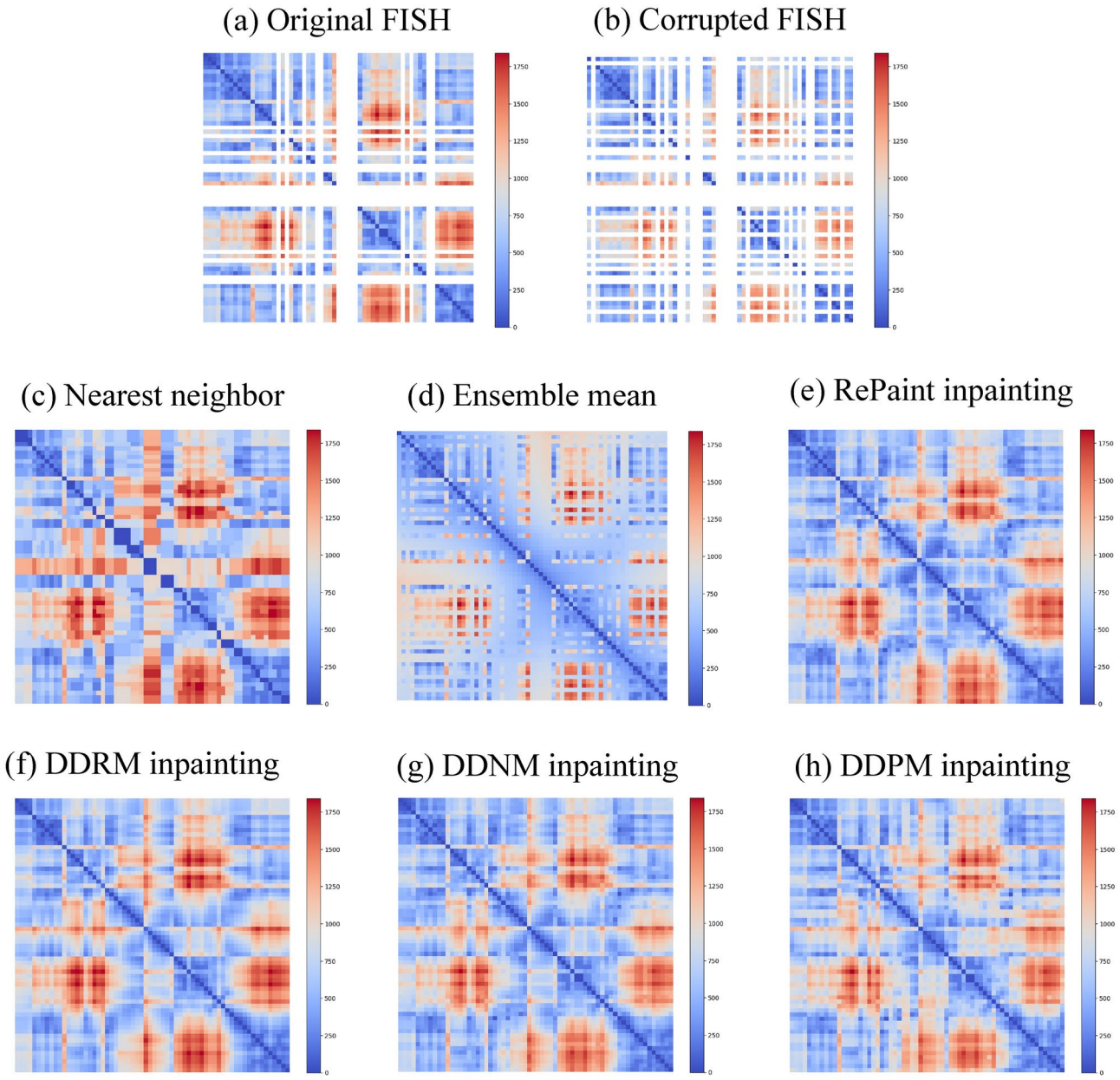

Fig. 5. Inpainting of chromosome distance matrices from a FISH experiment. (**a**) Original experimental matrix with 15 missing rows and columns. (**b**) Corrupted experimental matrix with 10 rows and columns additionally masked. The masking is needed in order to evaluate RMSE of various inpainting methods at the known (masked) values. The resulting sparsity is $\mu = 0.63$. (**c**)-(**h**) Inpainting methods, as indicated. For the ensemble mean (**d**) the average value is taken over 670 single cell distance matrices, where the corresponding matrix element is known. Diffusion-based methods (**e**)-(**h**) exploit the pre-trained diffusion model with the Hurst parameter $H = 1/3$. The colorbars show the range of pairwise distances between chromosomal loci in nm. The data is shown for cell 343, see Tables S2-S3 in the Supplementary Information.

Methods	Ens. mean	NN	DDPM	RePaint	DDNM	DDRM
RMSE, nm	147.4 ± 5.2	111.4 ± 3.2	97.2 ± 2.5	98.3 ± 2.7	85.1 ± 2.8	**84.2 ± 2.8**
Rank	0.752 ± 0.025	0.79 ± 0.03	0.79 ± 0.04	0.82 ± 0.04	0.82 ± 0.03	0.82 ± 0.03

Table 4. Reconstruction of chromatin (FISH) distance matrices. Average metrics over 670 single cells are shown.

it in numerical experiments. We further show that the diffusion-based inpainting not only learns the latent representation of the distance matrices, but also manages to properly reproduce the statistical features of the fBm ensemble (the memory exponent). Application of the pre-trained fBm benchmark for the microscopy-derived dataset of pairwise spatial distances between chromosomal segments demonstrates its superiority in reconstructing the missing distances over the standard approaches widely used in bioinformatics. We thus expect that other chromosomal datasets obtained in high-throughput experiments (such as Hi-C) that can be represented as matrices would benefit from the proposed fBm benchmark.

It should be noted that instance-based metrics for evaluating inpainting quality, such as RMSE, PSNR, and SSIM, are valid only when the inpainting ground truth exists (as is the case with the EDM dataset at low μ). In image domains, however, masking even a small area can yield multiple plausible inpainting results. For example, masking facial features such as eyes can result in various natural facial variations (e.g., differing eye colors). Therefore, conventional image datasets, such as CIFAR-10, ImageNet, or LAION, present ill-posed inpainting problems. In contrast, the EDM dataset we proposed uniquely facilitates studying the performance of generative models at the instance level. Additionally, it allows systematic comparisons between diffusion inpainting and database search results, highlighting the fundamental generalization capabilities of the diffusion models.

As the main limitation of our fBm benchmark, we stress out that it is not directly applicable to noisy EDMs, i.e. we assume that the known pairwise distances are exact. This should be seriously taken into account upon application of the benchmark to real datasets. Also the wild type chromosomes (*with* cohesin present in the cell) show a distinctively non-fractal statistics, thus requiring an adequate modification of the benchmark for the purposes of the data imputation. Other caveats are discussed in the flow of the paper.

Methods

We conducted all experiments on a workstation equipped with two NVIDIA RTX 4090 graphics cards and 256 GB of RAM. The DDPM model was parameterized using the UNet2DModel from the Diffusers library, and we trained the model for 100 epochs on datasets of generated fBm trajectories for dataset sizes ranging from 500 to 200,000 distance matrices (500, 1000, 2000, 5000, 20000 and 200000), see Table 3. For other experiments we used the diffusion model trained on the dataset of 200k distance matrices. For the inpaitning experiments we used 200 generation steps. The source code for the experiments is available at https://github.com/alobashev/fbm-inpainting-benchmark.

DDPM unconditional generation

Denoising Diffusion Probabilistic Models (DDPM) generate data by reversing a forward noising process[1,2]. The forward process incrementally adds Gaussian noise to an initial sample $x_0 \sim p_{\text{data}}$ from the distribution p_{data} over a sequence of $T \gg 1$ steps, following a variance schedule denoted by $\beta_0 = 0;\ 0 < \beta_1, \beta_2, ..., \beta_T < 1$. When the schedule is properly configured and T is sufficiently large, the final noised sample becomes indistinguishable from pure Gaussian noise $\mathcal{N}(0, I)$. Thus, reversing the noise following the learned "pathway" enables the generation of a diverse array of new samples from the same distribution.

The forward process of the sequential noising can be characterized by the following sequence of conditional probability distributions:

$$q(x_t|x_{t-1}) = \mathcal{N}(x_t; \sqrt{1-\beta_t}x_{t-1}, \beta_t I), \qquad q(x_{1:T}|x_0) = \prod_{t=1}^{T} q(x_t|x_{t-1}). \tag{15}$$

Here $q(x_t|x_{t-1})$ represents the conditional probability of the next sample x_t for the previous sample x_{t-1}. At each step the mean of the sample is attenuated by a factor of $\sqrt{1-\beta_t} < 1$, bringing it closer to zero. The independence of the noising steps enables one to write down a closed-form expression for sampling at an arbitrary timestep *t*:

$$q(x_t|x_0) = \mathcal{N}\left(x_t; \sqrt{\bar{\alpha}_t}x_0, (1-\bar{\alpha}_t)I\right), \tag{16}$$

where we define $\alpha_t = 1 - \beta_t$ and the cumulative product $\bar{\alpha}_t = \prod_{i=1}^{t} \alpha_i$. The reverse process gradually denoises samples starting from $x_T \sim p(x_T)$ through a learned Markov chain:

$$p_\theta(x_{0:T}) = p(x_T) \prod_{t=1}^{T} p_\theta(x_{t-1}|x_t), \quad p(x_T) = \mathcal{N}(x_T; 0, I). \tag{17}$$

During the reverse process at each step we utilize a Gaussian probability distribution $p_\theta(x_{t-1}|t)$ characterized by the learned mean value $\mu_\theta(x_t, t)$ and a fixed variance

$$p_\theta(x_{t-1}|x_t) = \mathcal{N}\left(x_{t-1}; \mu_\theta(x_t, t), \sigma_t^2 I\right), \quad \sigma_t^2 = \beta_t. \tag{18}$$

Thus, the fundamental concept of learning during the noising stage is to numerically derive a sequence of mean values $\mu_\theta(x_t, t);\ t = T, T-1, ..., 1$ – an optimal path that connects a sample from the Gaussian distribution to a sample from the target distribution.

The training objective aims to minimize the variational lower bound (VLB) by utilizing the posterior distribution

$$q(x_{t-1}|x_t, x_0) = \mathcal{N}\left(x_{t-1}; \tilde{\mu}_t(x_t, x_0), \tilde{\beta}_t I\right), \tag{19}$$

where $\tilde{\mu}_t = \frac{\sqrt{\bar{\alpha}_{t-1}}\beta_t}{1-\bar{\alpha}_t} x_0 + \frac{\sqrt{\alpha_t}(1-\bar{\alpha}_{t-1})}{1-\bar{\alpha}_t} x_t$ and $\tilde{\beta}_t = \frac{1-\bar{\alpha}_{t-1}}{1-\bar{\alpha}_t}\beta_t$. Reparameterizing $x_t = \sqrt{\bar{\alpha}_t}x_0 + \sqrt{1-\bar{\alpha}_t}\epsilon$ leads to a simplified objective. We aim to approximate the noise addition at each step by defining a parametrized network Z_θ

$$L = \mathbb{E}_{x_0,\epsilon,t}\left[\|\epsilon - Z_\theta(x_t, t)\|^2\right], \tag{20}$$

where $\epsilon \sim \mathcal{N}(0, I)$ and t is uniformly sampled from $\{1, ..., T\}$. Once the network Z_θ is trained (i.e. the optimal parameters θ are found), the mean of the reverse process is computed as follows

$$\mu_\theta(x_t, t) = \frac{1}{\sqrt{\alpha_t}}\left(x_t - \frac{\beta_t}{\sqrt{1-\bar{\alpha}_t}} Z_\theta(x_t, t)\right). \tag{21}$$

In other words, the trained network $Z_\theta(x_t, t)$ at each step t provides the optimal direction for the shift relative to the sample x_t. The resulting vector $\mu_\theta(x_t, t)$ provides the mean for the subsequent sample x_{t-1} during the iterative generation process

$$x_{t-1} = \mu_\theta(x_t, t) + \sigma_t \varepsilon \quad \text{where } \varepsilon \sim \mathcal{N}(0, I). \tag{22}$$

In practice, we aim for faster sampling with fewer steps than those required for training (T). When shortening the Markov chain and utilizing only a subsequence S_i of the diffusion steps, the original schedule of variances must be adjusted to ensure that the marginal distributions remain equivalent, such that $q^{\text{new}}(x_i) = q(x_{S_i})$. From this condition, we can derive the betas for the new (shorter) diffusion process as follows

$$\beta_i^{\text{new}} = 1 - \frac{\bar{\alpha}_{S_i}}{\bar{\alpha}_{S_{i-1}}}. \tag{23}$$

The sampling process remains unchanged; it only necessitates a modification of the pretrained ϵ inputs to $\epsilon(x_i, S_i)$, where S_i corresponds to the respective timestep in the original chain. For all our experiments, we set the number of sampling iterations to 150.

Require: None

1: $\mathbf{x}_T \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$
2: **for** $t = T$ **to** 1 **do**
3: $\quad \boldsymbol{\varepsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ **if** $t > 1$, **else** $\boldsymbol{\varepsilon} = \mathbf{0}$
4: $\quad \mathbf{x}_{t-1} = \frac{1}{\sqrt{\alpha_t}}\left(\mathbf{x}_t - Z_\theta(\mathbf{x}_t, t)\frac{\beta_t}{\sqrt{1-\bar{\alpha}_t}}\right) + \sigma_t \boldsymbol{\varepsilon}$
5: **return** $\mathbf{x}_0$

Algorithm 1. Reverse Diffusion Process of DDPM Unconditional Generation

Also note that in the reverse diffusion process, the noise term $\boldsymbol{\epsilon}$ is sampled from the normal distribution $\mathcal{N}(0, \mathrm{I})$ for $t > 1$. However, for the final step ($t = 1$), the noise term is set to 0 to ensure deterministic sampling at the end of the process.

DDPM conditional generation (inpainting)

Given the mask B and the initial image x_0, the pre-trained DDPM allows us to address the inpainting problem in the following manner

$$x_{t-1}^{\text{unknown}} \sim \mathcal{N}(\mu_\theta(x_t, t), \Sigma_\theta(x_t, t)) \tag{24}$$

$$x_{t-1}^{\text{known}} \sim \mathcal{N}(\sqrt{\bar{\alpha}_{t-1}}x_0, (1-\bar{\alpha}_{t-1})I) \tag{25}$$

$$x_{t-1} = B \odot x_{t-1}^{\text{known}} + (I - B) \odot x_{t-1}^{\text{unknown}}. \tag{26}$$

At each step the known pixels are obtained from the initial image $y = B \odot x_0$, noised at the corresponding level $\bar{\alpha}_{t-1}$. The unknown pixels are sampled from the previous iteration x_t using the pre-trained unconditional model $\mu_\theta(x_t, t)$. The two results are then merged to form the new image x_{t-1}.

Require: The masked image $\mathbf{y}$, the mask $\mathbf{B}$

$\mathbf{x}_T \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$
for $t = T$ **to** 1 **do**
  $\varepsilon_1, \varepsilon_2 \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ **if** $t > 1$, **else** $\varepsilon_1, \varepsilon_2 = \mathbf{0}$
  $\mathbf{y}_{t-1} = \sqrt{\bar{\alpha}_{t-1}}\mathbf{y} + \sqrt{1-\bar{\alpha}_{t-1}}\varepsilon_1$
  $\mathbf{x}_{t-1} = \frac{1}{\sqrt{\alpha_t}}\left(\mathbf{x}_t - Z_\theta(\mathbf{x}_t, t)\frac{\beta_t}{\sqrt{1-\bar{\alpha}_t}}\right) + \sigma_t \varepsilon_2$
  $\mathbf{x}_{t-1} = \mathbf{y}_{t-1} + (\mathbf{I} - \mathbf{B})\mathbf{x}_{t-1}$
return $\mathbf{x}_0$

Algorithm 2. Reverse Diffusion Process of DDPM Inpainting

RePaint

The DDPM inpainting has recently been adapted into various variants, one of which is known as Repaint[11]. This variant employs a "back and forward" strategy to refine the inpainting results. Unlike the traditional DDPM approach, which progresses through a single pass of forward and backward steps (effectively a single loop), Repaint utilizes a method in which the reverse diffusion process is executed multiple times (n_t), or in "loops," to achieve higher fidelity results.

Require: The masked image $\mathbf{y}$, the mask $\mathbf{B}$

$\mathbf{x}_T \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$
for $t = T$ **to** 1 **do**
  for $n = 1$ **to** n_t **do**
    $\varepsilon_1, \varepsilon_2 \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ **if** $t > 1$, **else** $\varepsilon_1, \varepsilon_2 = \mathbf{0}$
    $\mathbf{y}_{t-1} = \sqrt{\bar{\alpha}_{t-1}}\mathbf{y} + \sqrt{1-\bar{\alpha}_{t-1}}\varepsilon_1$
    $\mathbf{x}_{t-1} = \frac{1}{\sqrt{\alpha_t}}\left(\mathbf{x}_t - Z_\theta(\mathbf{x}_t, t)\frac{\beta_t}{\sqrt{1-\bar{\alpha}_t}}\right) + \sigma_t \varepsilon_2$
    $\mathbf{x}_{t-1} = \mathbf{y}_{t-1} + (\mathbf{I} - \mathbf{B})\mathbf{x}_{t-1}$
    $\mathbf{x}_t = \sqrt{1-\beta_t}\mathbf{x}_{t-1} + \sqrt{\beta_t}\varepsilon_2$
return $\mathbf{x}_0$

Algorithm 3. Reverse Diffusion Process of RePaint

In principle, n_t can vary depending on the denoising step, *t*. It is important to note that DDPM is equivalent to RePaint when the number of loops is set to one, i.e. $n_t = 1$.

DDRM

The forward diffusion process defined by DDRM[13] is

$$\mathrm{x}_t = \mathrm{x}_0 + \sigma_t \boldsymbol{\epsilon}, \quad \boldsymbol{\epsilon} \sim \mathcal{N}(0, \mathrm{I}) \tag{27}$$

The original reverse diffusion process of DDRM is based on DDIM[57], which is

$$\mathrm{x}_{t-1} = \mathrm{x}_0 + \sqrt{1-\eta^2}\sigma_{t-1}\frac{\mathrm{x}_t - \mathrm{x}_0}{\sigma_t} + \eta\sigma_{t-1}\boldsymbol{\epsilon} \tag{28}$$

Require: The degraded image $\mathbf{y}$ with noise level $\sigma_{\mathbf{y}}$, the operator $\mathbf{B} = \mathbf{U}\Sigma\mathbf{V}^\top$, where $\mathbf{B} \in \mathbb{R}^{d \times D}$

$\mathbf{x}_T \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$
$\bar{\mathbf{y}} = \Sigma^\dagger \mathbf{U}^\top \mathbf{y}$
for $t = T$ **to** 1 **do**
  $\varepsilon \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ **if** $t > 1$ **else** $\varepsilon = \mathbf{0}$
  $\bar{\mathbf{x}}_{0|t} = \mathbf{V}^\top \frac{1}{\sqrt{\bar{\alpha}_t}} \left(\mathbf{x}_t - Z_\theta(\mathbf{x}_t, t)\sqrt{1-\bar{\alpha}_t}\right)$
  for $i = 1$ **to** D **do**
    if $s_i = 0$ **then**
      $\bar{\mathbf{x}}_{t-1}^{(i)} = \bar{\mathbf{x}}_{0|t}^{(i)} + \sqrt{1-\eta^2}\sigma_{t-1}\frac{\bar{\mathbf{x}}_t^{(i)} - \bar{\mathbf{x}}_{0|t}^{(i)}}{\sigma_t} + \eta\sigma_{t-1}\varepsilon^{(i)}$
    else if $\sigma_{t-1} < \frac{\sigma_{\mathbf{y}}}{s_i}$ **then**
      $\bar{\mathbf{x}}_{t-1}^{(i)} = \bar{\mathbf{x}}_{0|t}^{(i)} + \sqrt{1-\eta^2}\sigma_{t-1}\frac{\bar{\mathbf{y}}^{(i)} - \bar{\mathbf{x}}_{0|t}^{(i)}}{\sigma_{\mathbf{y}}/s_i} + \eta\sigma_{t-1}\varepsilon^{(i)}$
    else if $\sigma_{t-1} \geq \frac{\sigma_{\mathbf{y}}}{s_i}$ **then**
      $\bar{\mathbf{x}}_{t-1}^{(i)} = \bar{\mathbf{y}}^{(i)} + \sqrt{\sigma_{t-1}^2 - \frac{\sigma_{\mathbf{y}}^2}{s_i^2}}\varepsilon^{(i)}$
  $\mathbf{x}_{t-1} = \mathbf{V}\bar{\mathbf{x}}_{t-1}$
return $\mathbf{x}_0$

Algorithm 4. Reverse Diffusion Process of DDRM

DDNM

The forward and backward process in DDNM[12] is similar to the DDRM.

Require: The degraded image $\mathbf{y}$, the degradation operator $\mathbf{B}$ (mask), and its pseudo-inverse $\mathbf{B}^\dagger$.

$\mathbf{x}_T \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$
for $t = T$ **to** 1 **do**
  $\varepsilon \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ **if** $t > 1$ **else** $\varepsilon = \mathbf{0}$
  $\mathbf{x}_{0|t} = \frac{1}{\sqrt{\bar{\alpha}_t}} \left(\mathbf{x}_t - Z_\theta(\mathbf{x}_t, t)\sqrt{1-\bar{\alpha}_t}\right)$
  $\hat{\mathbf{x}}_{0|t} = \mathbf{x}_{0|t} - \mathbf{B}^\dagger(\mathbf{B}\mathbf{x}_{0|t} - \mathbf{y})$
  $\mathbf{x}_{t-1} = \frac{\sqrt{\bar{\alpha}_{t-1}}\beta_t}{1-\bar{\alpha}_t}\hat{\mathbf{x}}_{0|t} + \frac{\sqrt{\alpha_t}(1-\bar{\alpha}_{t-1})}{1-\bar{\alpha}_t}\mathbf{x}_t + \sigma_t\varepsilon$
return $\mathbf{x}_0$

Algorithm 5. Reverse Diffusion Process of DDNM

The DDNM adjustment ensures that the predicted image $\mathrm{x}_{0|t}$ respects the null space of the degradation operator B. To incorporate the Repaint-like steps, a time schedule is introduced as an array of tuples, where each tuple represents a pair of timesteps. This iterative process alternates between forward and backward steps, refining the image while preserving the known regions defined by B.

Require: The degraded image $\mathbf{y}$, the degradation operator $\mathbf{A}$ (mask), and its pseudo-inverse $\mathbf{A}^\dagger$, time schedule for repaint iterations.

$\mathbf{x}_T \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ ▷ Initialize noisy matrix
for (i, j) in time schedule **do** ▷ Repaint-like iteration
  if $j < i$ **then** ▷ Forward step
    $\mathbf{x}_{0|t} = \frac{1}{\sqrt{\bar{\alpha}_t}}\left(\mathbf{x}_t - Z_\theta(\mathbf{x}_t, t)\sqrt{1-\bar{\alpha}_t}\right)$ ▷ Predict denoised image
    $\hat{\mathbf{x}}_{0|t} = \mathbf{x}_{0|t} - \mathbf{B}^\dagger(\mathbf{B}\mathbf{x}_{0|t} - \mathbf{y})$ ▷ Null-space adjustment
    $\mathbf{x}_{t-1} = \frac{\sqrt{\bar{\alpha}_{t-1}}\beta_t}{1-\bar{\alpha}_t}\hat{\mathbf{x}}_{0|t} + \frac{\sqrt{\alpha_t}(1-\bar{\alpha}_{t-1})}{1-\bar{\alpha}_t}\mathbf{x}_t + \sigma_t\varepsilon$ ▷ Update noisy image
  else ▷ Backward step
    $\mathbf{x}_{t-1} = \sqrt{\bar{\alpha}_{t-1}}\mathbf{x}_{0|t} + \sqrt{1-\bar{\alpha}_{t-1}}\varepsilon$
$\mathbf{x}_0 = \mathbf{x}_t - \mathbf{A}^\dagger(\mathbf{A}\mathbf{x}_t - \mathbf{y})$
return $\mathbf{x}_0$

Algorithm 6. Reverse Diffusion Process of DDNM with Repaint-like Iteration

Database Search for Inpainting

Existence of the ground truth of the reconstruction and the respective threshold, below which the solution is unique, allows us to directly compare the reconstruction error of a generative diffusion model with the other methods. In particular, in what follows, we will test the performance of the database search, i.e. where the missing distances are extracted from the most similar matrix from the database evaluated at known distances.

It has been demonstrated that modern text-to-image diffusion models, such as StableDiffusion[17], partially memorize samples from the training dataset[14,15]. This leads to the question of whether diffusion models could function as an approximate database search.

In the context of inpainting, the database search approach leverages a pre-existing database of distance matrices of ensemble of trajectories. The error ε_i, characterizing the discrepancies between the corrupted matrix and each matrix i in the database, is computed as

$$\varepsilon_i = ||B \odot (A_i^{(DB)} - \tilde{A})||_F^2, \tag{29}$$

where $\tilde{A}$ is the partially known distance matrix, B is the binary mask of known distances, and $A_i^{(DB)}$ is an ensemble of complete EDMs of real trajectories from the database.

Let us denote the index of the matrix from the database that provides the minimum to the error Eq. 29 by $i^* = \text{argmin}(\varepsilon_i)$. Then the reconstruction $\hat{A}$ is obtained by integrating information from the original corrupted matrix and the closest match from the database and is given by

$$\hat{A} = B \odot \tilde{A} + (1 - B) \odot A_{i^*}^{(DB)}. \tag{30}$$

Note, that if $\tilde{A}$ belongs to the database $A^{(DB)}$ then this procedure would would give a reconstruction with zero error (if the EDM completion is unique). In other words, this inpainting procedure ideally over-fits the training data.

FISTA for low-rank distance matrix completion

Another approach to fill the unknown distances is to apply convex relaxation, by combining the minimization objective $||B \odot (A - \tilde{A})||_F^2$ with the L1 norm on X's nuclear norm. This makes matrix completion a regularized least square problem

$$\min_A ||B \odot (A - \tilde{A})||_F^2 + \beta||A||_* \tag{31}$$

where $\tilde{A}$ is a partially known distance matrix, B is a binary mask which represents known distance matrix entities, and A is the matrix to be reconstructed, β is the regularization coefficient, and $||A||_*$ is the nuclear norm of the matrix A, equivalent to the sum of all A's eigenvalues.

There were proposed several methods to solve this problem, one of them is the Fast Iterative Shrinkage-Thresholding Algorithm (FISTA)[58]. FISTA operates by iteratively updating the solution via a proximal gradient method and employs Nesterov's acceleration to enhance the rate of convergence. FISTA serves as ground truth method for the solution of the low rank matrix completion problem in cases where solution is unique, i.e. when the partial graph defined by the mask B is rigid.

Define the singular value soft-thresholding operator as:

$$D_\beta(A) = U(\Sigma - \beta I)_+ V^T \tag{32}$$

where $A = U\Sigma V^T$ is the singular value decomposition of A and $(x)_+ := \max(x, 0)$. Then the FISTA update rule is given by:

$$t_{k+1} = \frac{1 + \sqrt{1 + 4t_k^2}}{2} \tag{33}$$

$$Z^{k+1} = A^{k+1} + \frac{t_k - 1}{t_{k+1}}(A^{k+1} - A^k) \tag{34}$$

$$A^{k+1} = D_\beta(\tilde{A} \odot B + (1 - B) \odot Z^{k+1}) \tag{35}$$

and the initial approximation A^0 is the matrix $\tilde{A}$ with unknown elements filled with zeros. The stopping criterion for the iterative process is based on the ratio of the loss from Eq. 31 over two consecutive iterations.

Inpainting using trajectory optimization

The EDM completion problem can be tackled from the trajectory optimization perspective. The objective function to be minimized is the mean squared error (MSE) between the original and reconstructed distance matrices, computed over the known elements of the corrupted distance matrix. The loss function L over the trajectory $x = \{x\}_{i=1,\ldots,N}$ can be defined as:

$$L(x) = ||B \odot (A(x) - \tilde{A})||_F^2 = \tag{36}$$

$$= \sum_{||x_i - x_j|| \text{ known}} (a_{ij}(x) - \tilde{a}_{ij})^2. \tag{37}$$

The optimization process is carried out in *n* steps. At each step, a reconstructed distance matrix A is obtained by calculating the pairwise Euclidean distance between the points in the current trajectory. The trajectory x is updated iteratively via the Adam optimizer to minimize the loss function L. The final output is the reconstructed distance matrix

$$\hat{A} = A(x^*), \quad x^* = \arg\min L(x), \tag{38}$$

which is expected to approximate the original distance matrix $\tilde{A}$ under the given constraints.

A greedy algorithm for the graph rigidity

To better understand the idea of the algorithm, see Figure S1. At the first step we identify the largest clique (red nodes). For the clique we can reconstruct the full set of coordinates, as all cliques are universally rigid. At the next step we are looking for a vertex outside of the clique that has the maximum number of links to the clique, but not less than 4 (the blue node in Figure S1). If there is no such a vertex, the algorithm terminates and we conclude that the graph is not rigid. If it exists, we can determine the 3D coordinates of this vertex given the known coordinates of the vertices in the clique. In this case, we effectively know all the pairwise distances between the new blue vertex and all the red vertices in the clique, i.e. the number of vertices in the clique effectively increases by 1. We continue adding new vertices to the growing clique (green, orange) by repeating these steps, until all graph vertices join the clique. If the algorithm terminates before that, the graph is said to be not rigid.

In the following listing we provide the pseudo-code of the algorithm checking for the rigidity of any given graph. If the graph passes this check, it is rigid and allows for the unique immersion in the 3D metric space up to the distance-preserving transformations.

Require: M ▷ M is the binary mask representing known elements with 0 and unknowns with 1
$N \leftarrow \text{length}(M)$ ▷ N is the number of vertices in the graph
$S_{max} \leftarrow 0$ ▷ Initializes the size of the largest clique found to 0
$B \leftarrow \emptyset$ ▷ B is the best set of vertices found so far, initially empty
for $i \leftarrow 1$ to N **do**
  $C \leftarrow \{i\}$
  for $j \leftarrow 1$ to N **do**
    if $i \neq j$ **and** $M[i, j] = 0$ **and** $\forall k \in C, M[k, j] = 0$ **then**
      $C \leftarrow C \cup \{j\}$
  if $\text{length}(C) > S_{max}$ **then**
    $S_{max} \leftarrow \text{length}(C)$
    $B \leftarrow C$
while $\text{length}(B) < N$ **do**
  $K_{max} \leftarrow -1$
  $P_{best} \leftarrow$ null
  for $i \leftarrow 1$ to N **do**
    if $i \notin B$ **then**
      $K_{count} \leftarrow \sum_{j \in B}[M[i, j] = 0]$
      if $K_{count} > K_{max}$ **then**
        $K_{max} \leftarrow K_{count}$
        $P_{best} \leftarrow i$
  if $K_{max} < 4$ **then**
    return False
  else
    $B \leftarrow B \cup \{P_{best}\}$
    $M[P_{best}, :] \leftarrow 0$
    $M[:, P_{best}] \leftarrow 0$
return True

Algorithm 7. Rigidity test

Data availability

The source code for the experiments is available at https://github.com/alobashev/fbm-inpainting-benchmark.

Received: 14 August 2024; Accepted: 8 April 2025
Published online: 31 May 2025

References

1. Ho, J., Jain, A. & Abbeel, P. Denoising diffusion probabilistic models. *Advances in neural information processing systems* **33**, 6840–6851 (2020).
2. Song, Y. *et al.* Score-based generative modeling through stochastic differential equations. *arXiv preprint* arXiv:2011.13456 (2020).
3. Sohl-Dickstein, J., Weiss, E., Maheswaranathan, N. & Ganguli, S. Deep unsupervised learning using nonequilibrium thermodynamics. In *International conference on machine learning*, 2256–2265 (2015).
4. Dhariwal, P. & Nichol, A. Diffusion models beat gans on image synthesis. In *Advances in Neural Information Processing Systems*, 8780–8794 (2021).
5. Vahdat, A. & Kautz, J. NVAE: A deep hierarchical variational autoencoder. *Advances in neural information processing systems* **33**, 19667–19679 (2020).
6. Diederik, P. & Welling, M. Auto-encoding variational bayes. In *International Conference on Learning Representations*, 8780–8794 (2014).
7. Karras, T. *et al.* Analyzing and improving the image quality of stylegan. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 8110–8119 (2020).
8. Brock, A. & Hernán, M. Large scale GAN training for high fidelity natural image synthesis (2018). ArXiv, https://arxiv.org/abs/1809.11096.
9. Goodfellow, I. *et al.* Generative adversarial nets. In *Advances in neural information processing systems* (2014).
10. Ramesh, A., Dhariwal, P., Nichol, A., Chu, C. & Chen, M. Hierarchical text-conditional image generation with clip latents (2022). ArXiv, https://arxiv.org/abs/2204.06125.
11. Lugmayr, A. *et al.* Repaint: Inpainting using denoising diffusion probabilistic models. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* 11461–11471 (2022).
12. Wang, Y., Yu, J. & Zhang, J. Zero-shot image restoration using denoising diffusion null-space model. *The Eleventh International Conference on Learning Representations* (2023).
13. Kawar, B., Elad, M., Ermon, S. & Song, J. Denoising diffusion restoration models. *Advances in Neural Information Processing Systems* **35**, 23593–23606 (2022).
14. Carlini, N. *et al.* Extracting training data from diffusion models. In *32nd USENIX Security Symposium (USENIX Security 23)*, 5253–5270 (2023).
15. Somepalli, G., Singla, V., Goldblum, M., Geiping, J. & Goldstein, T. Diffusion art or digital forgery? investigating data replication in diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 6048–6058 (2023).
16. Saharia, C. et al. Photorealistic text-to-image diffusion models with deep language understanding. *Advances in Neural Information Processing Systems* **35**, 36479–36494 (2022).
17. Rombach, R., Blattmann, A., Lorenz, D., Esser, P. & Ommer, B. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 10684–10695 (2022).
18. Metzler, R. & Klafter, J. The random walk's guide to anomalous diffusion: a fractional dynamics approach. *Physics Reports* **339**, 1–77 (2000).
19. Watson, J. et al. De novo design of protein structure and function with rfdiffusion. *Nature* **620**, 1089–1100 (2023).
20. Ingraham, J. *et al.* Illuminating protein space with a programmable generative model. *Nature* 1–9 (2023).

21. Wang, Y. & Cheng, J. HiCDiff: single-cell Hi-C data denoising with diffusion models. *bioRxiv* 2023–12 (2023).
22. Lieberman-Aiden, E. et al. Comprehensive mapping of long-range interactions reveals folding principles of the human genome. *Science* **326**, 289–293 (2009).
23. Bintu, B. *et al.* Super-resolution chromatin tracing reveals domains and cooperative interactions in single cells. *Science* **362**, eaau1783 (2018).
24. Imakaev, M. et al. Iterative correction of Hi-c data reveals hallmarks of chromosome organization. *Nature methods* **9**, 999–1003 (2012).
25. Galitsyna, A. A. & Gelfand, M. Single-cell Hi-c data analysis: safety in numbers. *Briefings in bioinformatics* **22**, bbab316 (2021).
26. Polovnikov, K. E. et al. Crumpled polymer with loops recapitulates key features of chromosome organization. *Physical Review X* **13**, 041029 (2023).
27. Polovnikov, K., Nechaev, S. & Tamm, M. V. Effective Hamiltonian of topologically stabilized polymer states. *Soft Matter* **14**, 6561–6570 (2018).
28. Polovnikov, K., Nechaev, S. & Tamm, M. V. Many-body contacts in fractal polymer chains and fractional brownian trajectories. *Physical Review E* **99**, 032501 (2019).
29. Polovnikov, K., Gherardi, M., Cosentino-Lagomarsino, M. & Tamm, M. Fractal folding and medium viscoelasticity contribute jointly to chromosome dynamics. *Physical Review Letters* **120**, 088101 (2018).
30. Tamm, M. V. & Polovnikov, K. *Dynamics of polymers: classic results and recent developments* (Advanced Problems of Phase Transition Theory. World Scientific, Order, Disorder and Criticality, 2018).
31. Polovnikov, K., Gorsky, A., Nechaev, S., Razin, V. & Ulianov, S. V. Non-backtracking walks reveal compartments in sparse chromatin interaction networks. *Scientific Reports* **10**, 1–11 (2020).
32. Ulianov, S. et al. Order and stochasticity in the folding of individual Drosophila genomes. *Nature communications* **12**, 41 (2021).
33. Onuchin, A. A., Chernizova, A. V., Lebedev, M. A. & Polovnikov, K. E. Communities in C. elegans connectome through the prism of non-backtracking walks. *Scientific Reports* **13**, 22923 (2023).
34. Astakhov, A. M., Nechaev, S. & Polovnikov, K. Statistical properties of a polymer globule formed during collapse with the irreversible coalescence of units. *Polymer Science, Series C* **60**, 25–36 (2018).
35. Nechaev, S. & Polovnikov, K. Rare-event statistics and modular invariance. *Physics-Uspekhi* **61**, 99 (2018).
36. Dokmanic, I., Parhizkar, R., Ranieri, J. & Vetterli, M. Euclidean distance matrices: essential theory, algorithms, and applications. *IEEE Signal Processing Magazine* **32**, 12–30 (2015).
37. Schoenberg, I. Remarks to Maurice Frechet's article sur la definition axiomatique d'une classe d'espace distances vectoriellement applicable sur l'espace de hilbert. *Annals of Mathematics* 724–732 (1935).
38. Young, G. & Householder, A. Discussion of a set of points in terms of their mutual distances. *Psychometrika* 19–22 (1938).
39. Menger, K. New foundation of Euclidean geometry. *American Journal of Mathematics* **53**, 721–745 (1931).
40. Krislock, N. & Wolkowicz, H. Euclidean distance matrices and applications. *Handbook on Semidefinite, Conic and Polynomial Optimization* 879-914 (2012).
41. Mucherino, A., Lavor, C., Liberti, L. & Maculan, N. *Distance Geometry: Theory, Methods, and Applications* (Springer Science & Business Media, New York, NY, 2012).
42. Liberti, L., Lavor, C., Maculan, N. & Mucherino, A. Euclidean Distance Geometry and Applications. *SIAM Rev.* **56**, 3–69 (2014).
43. Mead, A. Review of the development of multidimensional scaling methods. *Journal of the Royal Statistical Society: Series D (The Statistician)* **41**, 27–39 (1992).
44. Grone, R., Johnson, C. R., Sa, E. M. & Wolkowicz, H. Positive definite completions of partial Hermitian matrices. *Linear Algebra and Its Applications* **58**, 109–124 (1984).
45. Connelly, R. & Guest, S. D. *Frameworks, tensegrities, and symmetry* (Cambridge University Press, 2022).
46. Connelly, R. & Gortler, S. J. Iterative universal rigidity. *Discrete & Computational Geometry* **53**, 847–877 (2015).
47. Alfakih, A. On the universal rigidity of generic bar frameworks. *Contribution Disc. Math* **5**, 7–17 (2010).
48. Gortler, S. J. & Thurston, D. P. Characterizing the universal rigidity of generic frameworks. *Discrete and Computational Geometry* **51**, 1017–1036 (2014).
49. Linial, N., London, E. & Rabinovich, Y. The geometry of graphs and some of its algorithmic applications. *Combinatorica* **15**, 215–245 (1995).
50. Saxe, J. B. Embeddability of weighted graphs in k-space is strongly np-hard. *17th Allerton Conf. Commun. Control Comput.* (1979).
51. Mandelbrot, B. & Van Ness, J. W. Fractional Brownian motions, fractional noises and applications. *SIAM Rev.* **10**, 422–437 (1968).
52. Metzler, R., Jeon, J. H., Cherstvy, A. G. & Barkai, E. Anomalous diffusion models and their properties: non-stationarity, non-ergodicity, and ageing at the centenary of single particle tracking. *Physical Chemistry Chemical Physics* **16**, 24128–24164 (2014).
53. Bouchaud, J.-P. & Georges, A. Anomalous diffusion in disordered media: statistical mechanisms, models and physical applications. *Physics Reports* **195**, 127–293 (1990).
54. Davies, R. B. & Harte, D. S. Tests for Hurst effect. *Biometrika* **74**, 95–101 (1987).
55. Polovnikov, K. & Slavov, B. Topological and nontopological mechanisms of loop formation in chromosomes: Effects on the contact probability. *Physical Review E* **107**, 054135 (2023).
56. Slavov, B. & Polovnikov, K. Intrachain distances in a crumpled polymer with random loops. *JETP Letters* **118**, 208–214 (2023).
57. Song, J., Meng, C. & Ermon, S. Denoising diffusion implicit models. *arXiv preprint* arXiv:2010.02502 (2020).
58. Beck, A. & Teboulle, M. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM journal on imaging sciences* **2**, 183–202 (2009).

Acknowledgements

The authors are grateful to Artem Vasiliev, Mariano Barbieri and the members of the laboratory of Ralf Metzler for illuminating discussions. K.P. acknowledges the support of the Alexander von Humboldt Foundation. The work on inpainting of chromosomal matrices is supported by the Russian Science Foundation (Grant No. 25-13-00277).

Author contributions

K.P. conceived and conceptualized the study, A.L. and D.G. conducted the experiments, A.L. and K.P. analysed the data. All authors participated in writing the manuscript.

Funding

Open Access funding enabled and organized by Projekt DEAL.

Declarations

Competing interests
The authors declare no competing interests.

Additional information
Supplementary Information The online version contains supplementary material available at https://doi.org/10.1038/s41598-025-97893-5.

Correspondence and requests for materials should be addressed to K.P.

Reprints and permissions information is available at www.nature.com/reprints.

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.