Computer Methods in Applied Mechanics and Engineering 435 (2025) 117589

Contents lists available at ScienceDirect

Comput. Methods Appl. Mech. Engrg.

journal homepage: www.elsevier.com/locate/cma

Can diffusion models capture extreme event statistics?

Stamatis Stamatelopoulos, Themistoklis P. Sapsis *

Department of Mechanical Engineering, Massachusetts Institute of Technology, 77 Massachusetts Ave., Cambridge, MA 02139, United States of America

ARTICLE INFO

Keywords:
Diffusion modeling
Extreme events
Geometry

ABSTRACT

For many important problems it is essential to be able to accurately quantify the statistics of extremes for specific quantities of interest, such as extreme atmospheric weather events or ocean-related quantities. While there are many classical approaches to perform such modeling tasks, recent interest has been increasing in the usage of generative models trained on available data. Despite the sporadic success of such methods, it is not clear for what systems or datasets a system-agnostic generative AI tool is capable of generating previously 'unseen' extreme events in a manner that accurately extrapolates the tails for the observable of interest. Here, we propose an apriori criterion, which based on the geometry of the training dataset, it can predict whether a generative AI tool will be able to extrapolate the tails, i.e. generate previously unseen extreme events. The idea is to quantify whether existing extreme events lie in the interior of the dataset or its boundary. In the former case it is shown that generative AI tools can work in an 'interpolation' mode and generate new extreme events. On the other hand, if the topology of the dataset is such that extremes live in the boundary of the domain then the generative AI algorithm needs to operate in an extrapolation mode which does not lead to accurate results. We illustrate our findings on a specific class of Diffusion Models (DMs) called Denoising Diffusion Probabilistic Models (DDPMs) and we test on three datasets, a simple on-hyperball dataset following a Weibull distribution for the radii of the data points of dimensionality $2 \cdot 10^3$, a dataset sampled from the so-called Majda–McLaughlin–Tabak Wave Model (MMT), of dimensionality $8.1 \cdot 10^3$ and a dataset consisting of Lagrangian turbulence trajectories, of dimensionality $2 \cdot 10^3$.

1. Introduction

For a wide range of problems in engineering and science it is essential to be able to accurately quantify the extreme event statistics of specific quantities of interest. For example, in the area of ocean engineering, the appearance of rogue waves, which are uncharacteristically large waves for the given sea state with crest-to-trough heights exceeding two times the significant wave height [1–5], is a phenomenon whose precise statistical modeling is of fundamental importance as these waves can have catastrophic consequences on ships and other structures at sea [6–8].

There are multiple classical approaches for such modeling tasks, one of which is the employment of copulas [9–11], which are based on Sklar's theorem [12] and are able to identify the dependence between different univariate marginals. Another approach is multivariate exceedance models [13], which differ to copulas in that they learn the conditional model to identify the dependence structure between marginals, whereas the performance of copulas is based on choosing the appropriate copula family [14].

Recently, interest in using machinery imported from the area of unsupervised machine learning for the modeling of systems exhibiting extreme events is increasing [14–16]. Promising is the use of Generative Adversarial Networks (GANs) [17,18], where

[*] Corresponding author.
E-mail address: sapsis@mit.edu (T.P. Sapsis).

https://doi.org/10.1016/j.cma.2024.117589

Available online 3 December 2024

there are examples which are able to generate extremes of the desirable severity for rainfall phenomena over the United States [19]. We turn our focus on a different class of generative models, the so-called Diffusion Models, which are also a promising candidate with the appearance of models such as SwinRDM [20], a data-driven super-resolution weather forecasting model and FuXi-Extreme [16], another weather forecasting model optimized for the prediction of extreme surface variables. However, it is not clear under what conditions such unsupervised learning models are expected to work.

In this study we propose an a priori criterion, based on characteristics of the training dataset, for the efficacy of DDPMs to extend the tails of the training dataset, capturing the pertinent extreme statistics. In Section 2, we provide some preliminaries on the specific class of DMs that we consider as well as some background on the three datasets that we test our criterion on. In Section 3, we provide the criterion definition and details on its implementation algorithm. Finally in Section 5 we provide the key results of this study which are in support of the validity of the proposed criterion before we provide some concluding remarks in Section 6.

2. Background

Here we provide a brief overview of the diffusion model framework that has been used for this study, while for more details, the reader is referred to [15,21,22]. Let $\mathbf{V} \sim f_{\mathbf{V}}$ be a random variable where $f_{\mathbf{V}}$ is the underlying probability density of interest. A diffusion model trained on a dataset $\{\mathbf{v}_\alpha, \alpha \in I\}$ sampled according to $f_{\mathbf{V}}$, learns a random mapping $p(\cdot)$[1], such that for $\mathbf{Z} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$, $p(\mathbf{Z}) \sim f_{\mathbf{V}}$, thus allowing for the generation of samples on $f_{\mathbf{V}}$ through sampling $\mathcal{N}(\mathbf{0}, \mathbf{I})$.

Choosing n diffusion steps and a real sequence β_i, define a sequence of random variables $\mathbf{V}_i$ through their conditional PDFs,[2] as,

$$f_{\mathbf{V}_i|\mathbf{V}_{i-1}}(\cdot|\mathbf{v}_{i-1}) = \mathcal{N}(\sqrt{1-\beta_i}\mathbf{v}_{i-1}, \beta_i \mathbf{I}), \quad \text{where} \quad \mathbf{V} \equiv \mathbf{V}_0, \qquad i = 1, \dots, n \tag{1}$$

where the process of generating samples in $f_{\mathbf{V}_i|\mathbf{V}_{i-1}}$ from an initial sample $\mathbf{v}_0$ sampled on $f_{\mathbf{V}}$, is called the *forward diffusion process*. Notice that all variables $\mathbf{V}_i$ except $\mathbf{V}_0$ follow a multivariate normal distribution. Now, for $\mathbf{V}_i$ defined through (1), and $\bar{\alpha}_i = \prod_{j=0}^{i}(1-\beta_j)$, it can be shown that,

$$f_{\mathbf{V}_n|\mathbf{V}_0}(\cdot|\mathbf{v}_0) = \mathcal{N}(\sqrt{\bar{\alpha}_n}\mathbf{v}_0, (1-\bar{\alpha}_n)\mathbf{I}).$$

Letting $\mathbf{V}_i|_{\mathbf{v}_0}$ be the random variable with PDF $f_{\mathbf{V}_i|\mathbf{V}_0}(\cdot|\mathbf{v}_0)$, we can then write

$$\mathbf{V}_i|_{\mathbf{v}_0} = \sqrt{\bar{\alpha}_i}\mathbf{v}_0 + \sqrt{(1-\bar{\alpha}_i)}\epsilon \tag{2}$$

where $\epsilon \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$.

Next, the goal is for every i and $\mathbf{v}$ to learn the reverse conditional distribution $f_{\mathbf{V}_i|\mathbf{V}_{i+1}}(\cdot|\mathbf{v})$, which permits the construction of an appropriate reverse process $p(\cdot)$. Specifically, for $\mathbf{V}_i$ as in (1), let be given $f_{\mathbf{V}_i|\mathbf{V}_{i+1}}(\cdot|\mathbf{v})$ for all $i \in \{0, \dots, n-1\}$ and all $\mathbf{v}$. Then define $p(\mathbf{v})$ by first sampling $f_{\mathbf{V}_{n-1}|\mathbf{V}_n}(\cdot|\mathbf{v})$ to retrieve $\mathbf{v}_{n-1}$. Then repeat for $f_{\mathbf{V}_{n-2}|\mathbf{V}_{n-1}}(\cdot|\mathbf{v}_{n-1})$ to retrieve $\mathbf{v}_{n-2}$ and so on, until $\mathbf{v}_0$, which is defined as the output of the *reverse diffusion process* $p(\cdot)$. It is then the case that $p(\mathbf{V}_n) \sim f_{\mathbf{V}_0}$. It can also be shown that for small β_n, $f_{\mathbf{V}_{i+1}|\mathbf{V}_i}$ will have a similar form to $f_{\mathbf{V}_i|\mathbf{V}_{i+1}}$ [22]. That is, $f_{\mathbf{V}_{i+1}|\mathbf{V}_i}$ will approximately follow a Gaussian distribution, which implies that a diffusion model with parameters θ needs only to learn the mean $\mu_\theta(i, \mathbf{v})$ and covariance matrix $\Sigma_\theta(i, \mathbf{v})$ for all i and $\mathbf{v}$. To proceed, we denote with g the learned distributions,

$$g_{\mathbf{V}_i|\mathbf{V}_{i+1}}(\cdot|\mathbf{v}) = \mathcal{N}(\mu_\theta(i, \mathbf{v}), \Sigma_\theta(i, \mathbf{v})) \tag{3}$$

so that the learned PDF of $\mathbf{V}_0$ is,

$$g_{\mathbf{V}_0}(\mathbf{v}) = \int g_{\mathbf{V}_0,\dots,\mathbf{V}_n}(\mathbf{v}, \mathbf{v}_1, \dots, \mathbf{v}_n) d\mathbf{V}_{1:n}. \tag{4}$$

We then attempt to minimize, with respect to θ, the cross entropy, L_{CE}, between the actual PDF of $\mathbf{V}_0$, $f_{\mathbf{V}_0}$ and the recovered PDF of $\mathbf{V}_0$, $g_{\mathbf{V}_0}$, which is defined as,

$$L_{CE} = -\mathbb{E}_{f_{\mathbf{V}_0}} \log(g_{\mathbf{V}_0}(\mathbf{V}_0)). \tag{5}$$

In [22], an upper bound L_{VLB} for L_{CE} is given, which is the minimization target. It can be shown that the minimization of L_{VLB} by θ is equivalent to minimizing

$$\mathbb{E}_{f_{\mathbf{V}_0}, \epsilon(\mathbf{v}_0, i)}\Big[\beta_i/(2(1-\beta_i)(1-\bar{\alpha}_i))\|\epsilon(\mathbf{v}_0, i) - \epsilon_\theta(\mathbf{v}_i, i)\|^2\Big], \qquad \text{for } i = 1, \dots, n \tag{6}$$

where $\epsilon(\mathbf{v}_0, i)$ is the specific noise sample used in (2) to retrieve $\mathbf{v}_i|_{\mathbf{v}_0}$. Finally, the above minimization target can be further simplified by not learning the variance associated with the reverse process, which preserves good performance [23], leading to the minimization targets used in this study,

$$L_i = \mathbb{E}_{f_{\mathbf{V}_0}, \epsilon(\mathbf{v}_0, i)}\Big[\|\epsilon(\mathbf{v}_0, i) - \epsilon_\theta(\mathbf{v}_i, i)\|^2\Big], \qquad \text{for } i = 1, \dots, n. \tag{7}$$

[1] I.e. even for the same deterministic input $\mathbf{v}$, $p(\mathbf{v})$ is a random variable

[2] Given (1) $f_{\mathbf{V}_i}$ can be recovered by integrating over $\mathbf{V}_{i-1}$

3. The criterion

We propose a criterion for the efficacy of diffusion models, as defined in Section 2, to extend the statistics of the dataset that they have been trained on. The criterion is related to the location of the extreme events of the pertinent statistic on the data manifold that the training dataset belongs to. Specifically, we conjecture that for a given quantity of interest, if the relevant extremes lie close to the boundary of the underlying manifold, then this inhibits the ability of the diffusion model to extend said statistic. We make the criterion statement more rigorous in the following definition,

Definition 3.1. Let $(\mathcal{X}, \langle\cdot,\cdot\rangle)$ be an inner product space of finite dimension N, and $\mathcal{M}$ an n-dimensional manifold with boundary $\partial\mathcal{M}$ embedded in $\mathcal{X}$. Then, let $\mathcal{D}_{\text{train}} \subset \mathcal{M}$ be a dataset of finite size, which will be used to train a diffusion model, and $\mathcal{D}_{\text{extreme}} \subset \mathcal{D}_{\text{train}}$ be a set of extreme events as defined by the relevant setting. Now, for any $p, q \in \mathcal{M}$, denote $\text{dist}(p, q)$ as the minimum on-manifold distance, induced from its ambient space, so that $\text{dist}(p, \partial\mathcal{M})$ is the distance between p and the boundary of $\mathcal{M}$. The criterion is then based on the conjecture that while diffusion models are able to learn $\mathcal{M}$ based on $\mathcal{D}_{\text{train}}$, they are unable to capture the boundary of $\partial\mathcal{M}$. The criterion then states that if $\mathcal{D}_{\text{extreme}}$ lies close to $\partial\mathcal{M}$ according to $\text{dist}(\cdot,\cdot)$, then a diffusion model trained on $\mathcal{D}_{\text{train}}$ will be unable to generate extreme events which extend the pertinent statistics in $\mathcal{D}_{\text{train}}$.

While for the on-hyperball dataset described in Section 4.1, $\text{dist}(p, \partial\mathcal{M})$ is easily deduced by the norm of $p \in \mathcal{M}$, in general, computing the distance of a point to the manifold boundary is not straightforward without further knowledge of $\mathcal{M}$. However, assuming that $\mathcal{M}$ does have a boundary, the algorithm presented in the remainder of this section allows us to gauge $\text{dist}(p, \partial\mathcal{M})$ between different points p, and is employed in this study.

3.1. Manifold boundary identification

In [24], the authors present BRIM, an algorithm for detecting *boundary points* of clusters, which they define as follows [25]; a boundary point p is an object that satisfies the following conditions,

1. It is within a dense region, R_1
2. There exists a region, R_2, near p with $\text{density}(R_1) \gg \text{density}(R_2)$ or $\text{density}(R_1) \ll \text{density}(R_2)$

In this section we demonstrate that BRIM also has potential for the detection of boundary points, as defined in the following context,

Definition 3.2. Let $(\mathcal{X}, \langle\cdot,\cdot\rangle)$ be an inner product space of finite dimension N, and $\mathcal{M}$ an n-dimensional manifold with boundary $\partial\mathcal{M}$ embedded in $\mathcal{X}$. Then, for $\mathcal{A} \subset \mathcal{M}$ a dataset of finite size, we say that a point $p \in \mathcal{A}$ is a boundary point of $\mathcal{A}$ according to the user defined parameter $\delta > 0$, if

$$\text{dist}(p, \partial\mathcal{M}) < \delta$$

where $\text{dist}(\cdot,\cdot)$ is the on-manifold distance, induced from its ambient space.

It should be noted that in [24], BRIM is tested only in 2-dimensional datasets in $\mathbb{R}^2$, so that $2 = n = N$. In what follows we present BRIM in the context of Definition 3.2 and argue that it is immediately applicable to higher-dimensional datasets.

The core idea in [24] is that points in the boundary of a cluster lie between regions of significantly different density. In terms of Definition 3.2, boundary points have the unique characteristic that $\mathcal{M}$ locally resembles a half-space $\mathbb{R}^n_{1/2}$ at ∂M. In this connection, for each point $p \in \mathcal{A}$, BRIM essentially first identifies the direction that the manifold $\mathcal{M}$ is relative to p, and then, in a user-specified neighborhood, compares how many points lie toward $\mathcal{M}$ versus away from $\mathcal{M}$. It follows that points whose neighborhood resembles $\mathbb{R}^n_{1/2}$, are able to be characterized by such a comparison. We proceed with an exact definition of the BRIM algorithm in the setting of Definition 3.2, starting with the standard definition of the δ-neighborhood of $p \in \mathcal{A}$, $N_\delta(p) \subseteq \mathcal{A}$ as,

$$N_\delta(p) = \{q \in \mathcal{A} : \ \|q - p\| < \delta\} \tag{8}$$

where $\|\cdot\| = \sqrt{\langle\cdot,\cdot\rangle}$ the norm induced by $\langle\cdot,\cdot\rangle$. Then, to identify the direction that $\mathcal{M}$ lies relative to p, define the *density attractor* of p, $p^*_\delta \in \mathcal{A}$, as

$$p^*_\delta = \underset{q \in N_\delta(p)}{\text{argmax}} |N_\delta(q)| \tag{9}$$

where $|\cdot|$ denotes cardinality. It is then assumed that $p^*_\delta - p$ points towards $\mathcal{M}$ when mounted at p. Next, consider the space normal to $p^*_\delta - p$,

$$\mathcal{N} = \{x \in \mathcal{X} : \ \langle x - p, p^*_\delta - p\rangle = 0\} \tag{10}$$

We say that a point in $N_\delta(p)$ lies towards $\mathcal{M}$, if it lies on *the same side* of $\mathcal{N}$ as p^*_δ does according to $\langle\cdot,\cdot\rangle$. Specifically, $x \in \mathcal{X}$ lies on the same side of $\mathcal{N}$ as p^*_δ if,

$$\langle x - p, p^*_\delta - p\rangle > 0$$

Equivalently, if x_{proj} is the projection of x onto $\mathcal{N}$,

$$\langle x - x_{\text{proj}}, p^*_\delta - p\rangle > 0$$

We can the proceed to the definition of the *boundary degree* of a point p,

Definition 3.3. Define the positive neighborhood of p, $N_\delta^+(p) \subseteq N_\delta(p)$, as those points which lie on the same side of $\mathcal{N}$ as does p_δ^*. Specifically, define $N_\delta^+(p) = \{q \in N_\delta(p) : \langle q - p, p_\delta^* - p\rangle \geq 0\}$, and similarly let the negative neighborhood of p be $N_\delta^-(p) = \{q \in N_\delta(p) : \langle q - p, p_\delta^* - p\rangle \leq 0\}$. Finally, let the boundary degree of p be

$$\partial_\delta p = \frac{|N_\delta^+(p)|}{|N_\delta^-(p)|}$$

Notice that $|N_\delta^-(p)| \neq 0$ and that $N_\delta^+(p) \cap N_\delta^-(p) \neq \emptyset$, since $p \in N_\delta^+(p) \cap N_\delta^-(p)$.

Now, BRIM comprises of the 4 steps outlined in Algorithm 1. Having all $\partial_\delta p$, the user may want to decide on a specific threshold value κ so that boundary points can be classified by $\partial_\delta p > \kappa$.

Algorithm 1

```
Given a dataset $\mathcal{A}$ and a user specified $\delta$ parameter
Compute $N_\delta(p)$ for all $p \in \mathcal{A}$ as in Equation (8)
For all $p \in \mathcal{A}$ compute $p_\delta^*$ as in Equation (9)
Compute all positive and negative $N_\delta^+(p), N_\delta^-(p)$ neighborhoods as in Definition 3.3
Compute all boundary degrees $\partial_\delta p$ as in Definition 3.3
```

Next, it is the case that the boundary degrees $\partial_\delta p$ identified by Algorithm 1, are *geometric* properties of the point-cloud $\mathcal{A}$, in the sense that they remain invariant to rigid transformations and embeddings of the following form,

$$\mathbb{R}^n \ni a_1 e_1 + \cdots + a_n e_n \to a_1 e_1 + \cdots + a_n e_n + 0e_{n+1} + \cdots + 0e_N \in \mathbb{R}^N.$$

To prove this, we begin by introducing the following convention; let $\partial_\delta p$, be the boundary degree of $p \in \mathcal{A}$ with respect to the dataset $\mathcal{A} \subset \mathcal{X}$ and $f : (\mathcal{X}, \langle\cdot,\cdot\rangle_\mathcal{X}) \to (\mathcal{Y}, \langle\cdot,\cdot\rangle_\mathcal{Y})$. By $\partial_\delta f(p)$ we refer to the boundary degree of $f(p)$ with respect to $f(\mathcal{A})$.

To proceed, recall that a transformation $R : \mathcal{X} \to \mathcal{X}$ is an orthogonal transformation on $(\mathcal{X}, \langle\cdot,\cdot\rangle)$, if

$$(\forall p, q \in \mathcal{X})(\langle R(u), R(v)\rangle = \langle u, v\rangle)$$

and $T : \mathcal{X} \to \mathcal{X}$ is a rigid transformation on $(\mathcal{X}, \langle\cdot,\cdot\rangle)$, if it can be written in the following form,

$$T(u) = R(u) + t$$

for some orthogonal transformation R and some $t \in \mathcal{X}$. Also, recall that if T is rigid, and $\|\cdot\|$ is the norm induced on $\mathcal{X}$ from its inner product, then

$$(\forall p, q \in \mathcal{X})(\|T(p) - T(q)\| = \|p - q\|) \tag{11}$$

It is then straightforward to prove that the boundary degree of $p \in \mathcal{A}$, is invariant to rigid transformations,

Proposition 3.1. *Let $\partial_\delta p$ be the boundary degree of $p \in \mathcal{A}$ with respect to the dataset $\mathcal{A} \subset \mathcal{X}$ and $T : \mathcal{X} \to \mathcal{X}$ be some rigid transformation on $\mathcal{X}$. Then, $\partial_\delta T(p) = \partial_\delta p$*

Proof. We begin by showing that $|N_\delta(p)| = |N_\delta(T(p))|$, for which the following two statements suffice,

$$q \in N_\delta(p) \iff T(q) \in N_\delta(T(p)) \quad \text{and} \quad q \neq q' \implies T(q) \neq T(q')$$

For the former we can write,

$$\begin{aligned} q \in N_\delta(p) &\iff \|q - p\| < \delta \\ &\underset{(11)}{\iff} \|T(q) - T(p)\| < \delta \\ &\iff T(q) \in N_\delta(T(p)) \end{aligned}$$

while for the latter, it is enough to notice that orthogonal transformations are invertible. Looking at Eq. (9), it then follows that

$$\Big(T(p)\Big)_\delta^* = T(p_\delta^*)$$

so that we can conclude the proof by writing

$$\begin{aligned} q \in N_\delta^+(p) &\iff \langle q - p, p_\delta^* - p\rangle \geq 0 \\ &\iff \langle T(q) - T(p), T(p_\delta^*) - T(p)\rangle \geq 0 \\ &\iff \langle T(q) - T(p), \Big(T(p)\Big)_\delta^* - T(p)\rangle \geq 0 \\ &\iff T(q) \in N_\delta^+(T(p)) \end{aligned}$$

We proceed similarly for showing that $|N_\delta^-(p)| = |N_\delta^-(T(p))|$. □

Similarly, we can prove that boundary degrees are also invariant to the elementary embedding of the relevant dataset on higher-dimensional spaces,

Proposition 3.2. *Consider a N-dimensional real inner product space $(\mathcal{X}_N, \langle\cdot,\cdot\rangle_N)$ and a basis $e_1,\dots,e_N$. Construct the $n < N$ dimensional space $(\mathcal{X}_n, \langle\cdot,\cdot\rangle_n)$ as follows: let $\mathcal{X}_n = \mathrm{span}(e_1,\dots,e_n)$ and define $\phi : \mathcal{X}_n \to \mathcal{X}_N$ as,*

$$\phi(a_1e_1 + \cdots + a_ne_n) = a_1e_1 + \cdots + a_ne_n + 0e_{n+1} + \cdots + 0e_N,$$

so that for $p, q \in \mathcal{X}_n$, $\langle p, q\rangle_n = \langle\phi(p), \phi(q)\rangle_N$. Then, for $\partial_\delta p$ the boundary degree of $p \in \mathcal{A}$ with respect to the dataset $\mathcal{A} \subset \mathcal{X}_n$, it is true that $\partial_\delta\phi(p) = \partial_\delta p$.

Proof. It is trivial to see that ϕ is injective. Then we can write,

$$\begin{aligned} q \in N_\delta(p) &\iff \langle q-p, q-p\rangle_n^{1/2} < \delta \\ &\iff \langle\phi(q)-\phi(p), \phi(q)-\phi(p)\rangle_N^{1/2} < \delta \\ &\iff \phi(q) \in N_\delta(\phi(p)) \end{aligned}$$

which follows from the linearity of ϕ. We can then proceed identically to Proposition 3.1. □

Again, BRIM works by looking at some neighborhood of size δ about every datapoint, and then identifies, for each point, the direction towards the core of the manifold (if the point is not near the boundary, any direction is equivalent) until, eventually, a boundary degree is assigned to every point indicating how close it is to the boundary. In the general case, tuning δ for any given dataset is not straightforward, with only the following statements available to us for the task in absence of other information about the underlying manifold,

1. For any $\delta > \delta_{\max}$ (the greatest distance between any two points) the output of BRIM is the same
2. For any $\delta < \delta_{\min}$ (the smallest distance between any two points) the output of BRIM is the same
3. If δ is too large, then any global features of the underlying geometry such as non-convexity, will be lost
4. If δ is too small, then not enough points will be included about every point, resulting in erroneous identification of the direction towards the interior of the manifold

Thus, in tuning δ, we want to appropriately chose some $\delta \in (\delta_{\min}, \delta_{\max})$. For the case of the hypersphere, where we know that the boundary degrees must be positively correlated with respect to norm of the data-points, we see a region closer to $\delta_{\min}$ which produces positive correlation, though performance reduces with increasing the dimension of the hyperball and reducing the number of points.

4. Datasets

We consider three datasets: (i) points distributed on a 2000-dimensional hyperball with their norm following a Weibull distribution (ii) spatial trajectories, each of dimension 8192, extracted from the MMT 1-dimensional wave turbulence model (iii) velocity time series, each of dimension 2000, extracted from a high-resolution numerical simulation of the 3D Navier–Stokes equations with large-scale isotropic forcing.

For the first dataset, it follows by design that extreme events of the norm statistic will lie close to the boundary of the underlying data manifold, which is a hyperball. Here we are interested in the statistic which computes the finite differences along the coordinates of each datapoint, for which it is still true that the points on the hyperball which exhibit the largest finite differences tend to lie close to the manifold boundary (see Section 5 for more details).

For the second and third datasets we observe the opposite, with points exhibiting the most extreme behavior being located in the interior of the dataset, as will be shown in Section 5 using Algorithm 1. In what follows, we provide more details on the three datasets and outline the sampling procedure followed for their generation.

4.1. On-hyperball distributions

Aiming to sample a distribution on the n-dimensional hyper-ball $\mathbf{B}^n(r)$ (25), such that extreme events are classified by their magnitude $\|\mathbf{x}\|$, a promising candidate is choosing a PDF of the form $f_{\mathbf{X}}(\mathbf{x}) = e^{-\lambda\|\mathbf{x}\|^2}$. This way, the probability of lying in regions of the hyperball away from its center will be significantly lower to those regions closer to it, thus classifying extremes as desired, i.e. occurring for large values of $\|\mathbf{x}\|$. Notice that this is precisely the form of a multivariate Gaussian distribution for an appropriate choice of parameters. Specifically, recalling the definition of an n-dimensional Gaussian vector $\mathbf{X} \sim \mathcal{N}(\mu, \Sigma)$, when Σ is positive definite, to be,

$$f_{\mathbf{X}}(\mathbf{x}) = \frac{\exp\Big(-0.5(\mathbf{x}-\mu)^T\Sigma^{-1}(\mathbf{x}-\mu)\Big)}{\sqrt{(2\pi)^n\det(\Sigma)}}$$

we can set $\mu = \mathbf{0}$ and $\Sigma = (2\pi)^{-1}\mathbf{I}$ to get,

$$\begin{aligned} f_{\mathbf{X}}(\mathbf{x}) &= \frac{\exp\Big(-0.5\mathbf{x}^T((2\pi)^{-1}\mathbf{I})^{-1}\mathbf{x}\Big)}{\sqrt{(2\pi)^n\det((2\pi)^{-1}\mathbf{I})}} \\ &= e^{-\pi\|\mathbf{x}\|^2} \end{aligned} \tag{12}$$

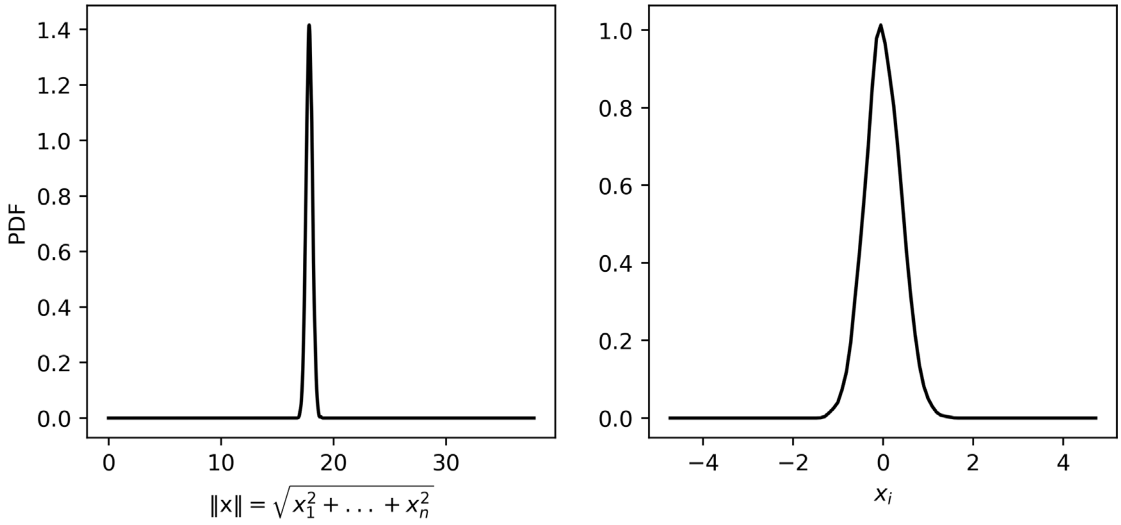

Fig. 1. Distribution of the radii (left) and the first coordinate (right) of (12). Notice that even though each projection of $\mathbf{X}$ naturally follows a Gaussian distribution, the magnitude of the entire vector is concentrated at $r \sim 17.8$. Specifically, $\|\mathbf{X}\|$ follows a χ-distribution in n degrees of freedom.

The PDFs of $\|\mathbf{x}\|$ and x_i for $n = 2000$ are plotted in Fig. 1, where we can see that the radii of the sampled points are concentrated at $r \sim \mathbb{E}[\|\mathbf{x}\|] = 17.83$ with $[\|\mathbf{x}\|_{\min}, \|\mathbf{x}\|_{\max}] = [16.57, 19.15]$.

Initially, it may seem that the datapoints $\mathbf{x}$ are distributed only in a tiny sliver of $\mathbf{B}^n(19.15)$. However, the ratio of the volume of the n-shell $\mathbf{Sh}^n(r_1; r_2)$ between radii $r_1 = 16.57 < 19.15 = r_2$ to the volume of $\mathbf{B}^n(19.15)$, is given by (28), (26) as,

$$\frac{\mathbf{Sh}^n(19.15, 16.57)}{\mathbf{B}^n(19.15)} = \frac{19.15^n - 16.57^n}{19.15^n} = 1 - 0.86^{2000} \sim 1$$

so that, in fact, essentially all of $\mathbf{B}^n(19.15)$ is covered by the distribution. However, the bulk of the points lying between 2 standard distributions of the mean of $\|\mathbf{x}\|$ in $[\mathbb{E}[\|\mathbf{x}\|] - 2\sigma, \mathbb{E}[\|\mathbf{x}\|] + 2\sigma] = [17.27, 18.4]$, do occupy only a fraction $\mathbf{B}^n(19.15)$,

$$\frac{\mathbf{Sh}^n(17.27, 18.4)}{\mathbf{B}^n(19.15)} \sim 0$$

Therefore, the manifold represented through (12) can be interpreted as the hypersphere $\mathbf{B}^n(\mathbb{E}[\|\mathbf{x}\|] + 2\sigma)$ along with some noise close to its boundary.

Alternatively we can proceed by directly choosing the distribution of the radii $\|\mathbf{x}\|$ as follows. Consider the n dimensional ball of radius r,

$$B_n = \{\mathbf{x} \in \mathbb{R}^n : \ \|\mathbf{x}\| < r\} \tag{13a}$$

and let

$$\mathbf{X} = Y\mathbf{Z}/\|\mathbf{Z}\|, \text{ where } Y \sim f_Y, \ \mathbf{Z} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}) \tag{13b}$$

Notice that $\|\mathbf{X}\|$ follows the distribution of $|Y|$. We choose Y to follow a Weibull distribution

$$f_Y(y; \lambda, a) = \begin{cases} \frac{a}{\lambda}\left(\frac{y}{\lambda}\right)^{a-1} e^{-(y/\lambda)^\alpha}, & y \geq 0 \\ 0, & y < 0, \end{cases} \tag{13c}$$

for $(a, \lambda) = (3/2, 1)$ the chosen values for the so-called shape and scale parameters.

Here we are interested in extreme event statistics for the finite difference,

$$(x_1, \dots, x_n) \to (\Delta_\tau x_1, \dots, \Delta_\tau x_{n-\tau}) \tag{14}$$

where $\Delta_\tau x_i = x_{i+\tau} - x_i$ and $\tau = 5$. Note that $\tau = 5$ is chosen arbitrarily and without any loss of generality. We note that the above difference is associated with pronounced tails, in contrast to the individual coordinates that follow a Gaussian distribution.

4.2. The Majda–McLaughlin–Tabak wave model

In the context of 1-dimensional wave turbulence, the following 2-parameter family of equations was proposed in [26],

$$i\partial_t u = |\partial_x|^a u + \lambda |\partial_x|^{-\beta/4}\left(\left||\partial_x|^{-\beta/4} u\right|^2 |\partial_x|^{-\beta/4} u\right) + iDu \qquad a > 0, \beta \in \mathbb{R} \tag{15}$$

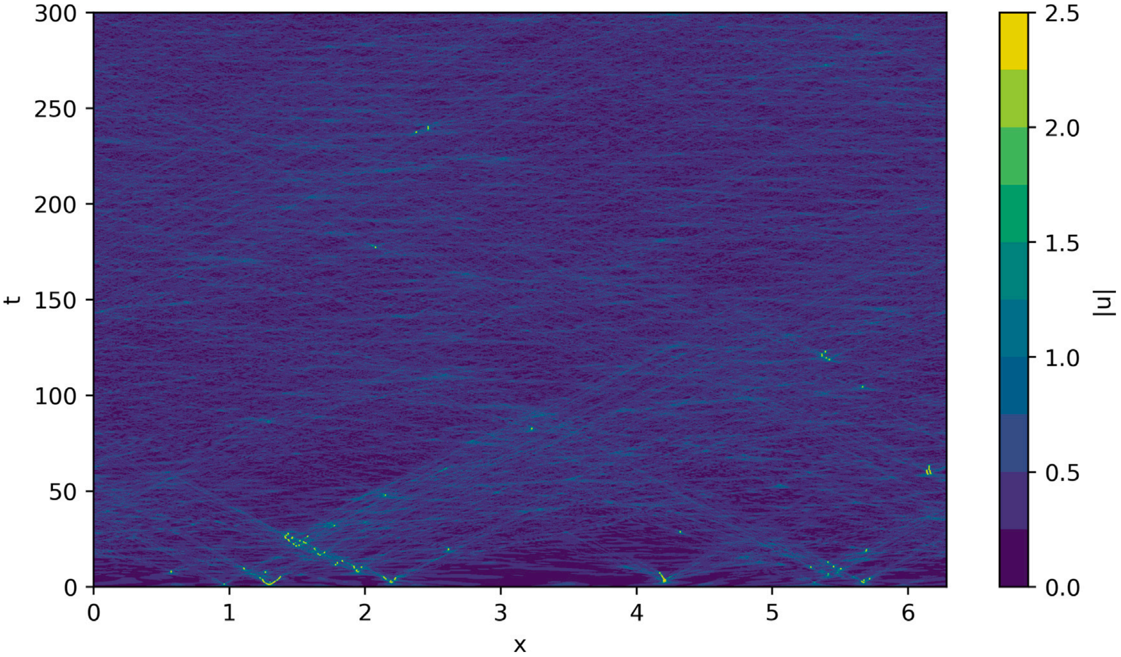

Fig. 2. Contour of a solution $|u(t,x)|$ of (18) for $\lambda = -4.0$ with timestep $h = 0.001$, over $x \in [0, 2\pi]$, $t \in [0, 300]$.

where $u(t,x) : \mathbb{R}^2 \to \mathbb{C}$ is interpreted as a stochastic process with t index, the pseudo-differential operator $|\partial_x|^a$ is defined through

$$\widehat{|\partial_x|^a u}(k) = |k|^a \hat{u}(k) \tag{16}$$

and the selective Laplacian operator Du defined as

$$\widehat{Du}(k) = \begin{cases} -(|k| - k^*)^2 \hat{u}(k) & |k| > k^* \\ 0 & |k| \leq k^*. \end{cases} \tag{17}$$

Du introduces dissipation for $|k| > k^*$, where k^* is a user-specified threshold. In analogy to [27], we chose the member $(a, \beta) = (1/2, 0)$ of (15),

$$i\partial_t u = |\partial_x|^{1/2} u + \lambda |u|^2 u + iDu \tag{18}$$

as then we retrieve the dispersion relation for deep waves $\omega^2 = |k|$. We solve (18), by employing ETD4RK [28], a 4th order Runge–Kutta exponential time differencing scheme, with more details given in Appendix B. In alignment with [27], we choose $\lambda = -4$ and solve (18) for $x \in [0, 2\pi]$ and $t \in [0, 300]$, with the initial conditions being the superposition of random phase harmonics. Specifically, we let

$$u(0,x) = \sum_{j=1}^{15} e^{i(jx + 2\pi\theta_{j,1})} + e^{i(-jx + 2\pi\theta_{j,2})} \tag{19}$$

where $\theta_{j,1}, \theta_{j,2}$ are independent identically distributed uniform random variables taking values in $[0, 1]$. For a specific realization of (19), we plot $|u(t,x)|$ in Fig. 2. It should be noted that $u(0,x)$ is periodic with period 2π, which implies that for any given realization of (19), the full range of initial conditions is given in $[0, 2\pi]$.

The above system is an example where extreme events 'live' in the interior of the domain. Specifically, as shown in [27] an extreme event may be formed if the combination of random initial phases is appropriate. Such a combination does not need to be the boundary of the dataset. For this reason, this is a good candidate dataset to extrapolate the tails using generative models.

As for the MMT dataset, the quantities of interest for this case are the finite differences Δu across each trajectory,

$$(|u_1|, \ldots, |u_n|) \to (\Delta_\tau |u_1|, \ldots, \Delta_\tau |u_{n-\tau}|) \tag{20}$$

where $\Delta_\tau |u_i| = |u_{i+\tau}| - |u_i|$ and $\tau = 5$. Again, $\tau = 5$ is chosen arbitrarily without any loss of generality and the quantity $\Delta_\tau |u|$, across the dataset, exhibits pronounced tails.

4.2.1. Generation and uncorrelated sampling of an ensemble of solutions

To proceed, we generate an ensemble of 300 solutions, and aim at sampling each member of this ensemble at various times t_i, so that for any one solution, all samples drawn from it are uncorrelated. Specifically, from each solution we extract $u(t_1, x), \ldots, u(t_N, x)$,

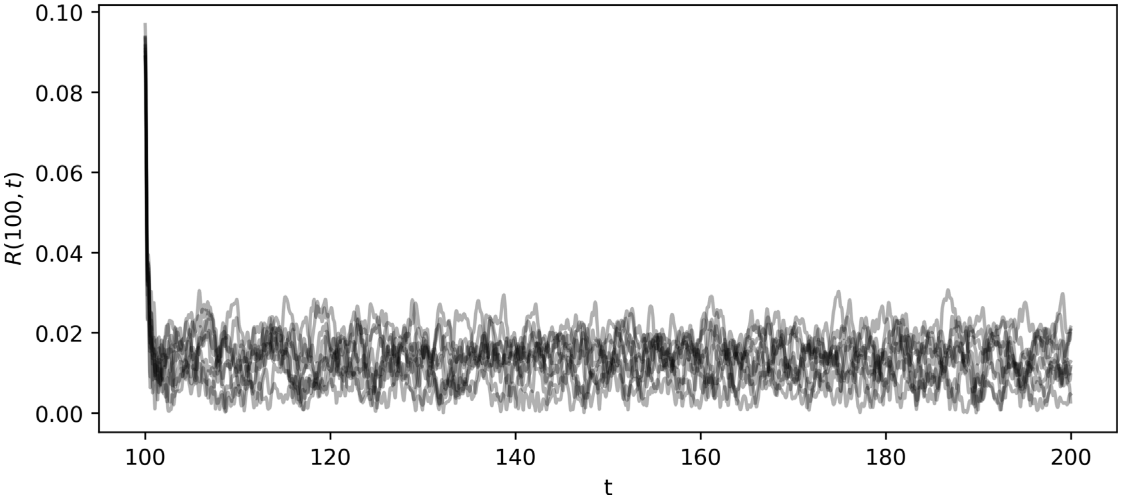

Fig. 3. Autocorrelation between $u(100, x)$ and $u(t, x)$, over an ensemble of 10 solutions of (18).

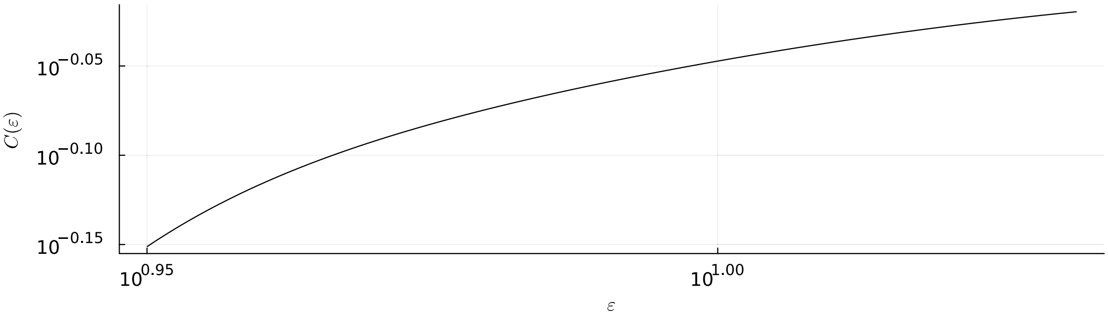

Fig. 4. Logarithmic plot of the correlation integral $C(\epsilon)$ [29] over the neighborhood parameter ϵ for the MMT dataset lying on $\mathcal{M}_{\mathrm{mmt}}$, computed via the Julia library `FractalDimensions.jl` [31].

such that any pair of random variables $u(t_i, x)$, $u(t_j, x)$ is uncorrelated. It should be noted that the discretization of the solution in x is of size 8192, which makes the dimensionality of the MMT dataset also 8192. In Fig. 3, we see the norm of the auto-correlation,

$$|R(t_i, t_j)| = |\mathbb{E}[u(t_i, x)\overline{u(t_j, x)}]| \tag{21}$$

for $t_i = 100$ and $t_j \in [100, 200]$, where $\bar{u}$ denotes complex conjugation. We note that the reason we choose the starting sampling time to be $t = 100$ is because even though the statistics of the $u(t, x)$ stochastic process never reach a statistical steady state, after $t = 100$, when a lot of the system energy provided by the initial conditions has been dissipated, the solution reaches a nearly statistical steady state [27]. To ensure that we have extracted uncorrelated samples we proceed as follows: (i) we pick a minimum t_{min} and maximum t_{max} distancing between samples (ii) for each solution in the ensemble we begin by picking a random $t \in [100, 100+t_{\mathrm{min}}-t_{\mathrm{max}}]$ and then taking steps of random length l, $t_{\mathrm{min}} < l < t_{\mathrm{max}}$, until we reach $t = 200$, at each step saving the relevant sample (iii) finally we compute the norm of the auto-correlation between all pairs of samples to ensure that it remains small enough. Notice that it is not enough for auto-correlation to be computed across consecutive samples as we have no guarantees about the stationarity of the random process $u(t, x)$. For the computed ensemble of size 300, we chose $t_{min} = 7.5$, retrieving 3755 samples, with $\max(|R(t_i, t_j)|) = 0.037$ which is in agreement with Fig. 3.

4.2.2. The dimension of the MMT data manifold

In this section we aim at providing some insight to the complexity of the MMT data manifold $\mathcal{M}_{\mathrm{mmt}}$, which is a subset of $\mathbb{R}^{8192}$. In contrast to the hyperball dataset, where the manifold is known, the $\mathcal{M}_{\mathrm{mmt}}$ is significantly more complicated to quantify, since it depends both on the numerical scheme used to approximate solutions $u(x, t)$ of (15) and also on the sampling scheme used to extract uncorrelated trajectories $u(t_i, x)$. However, viewing each point precisely as following some distribution on $\mathcal{M}_{\mathrm{mmt}}$, it is straightforward to calculate the correlation dimension for the relevant dataset [29]. Applying the Grassberger–Procaccia algorithm [30] to estimate it, through the Julia library `FractalDimensions.jl` [31], we arrive at the conclusion that $\dim(\mathcal{M}_{\mathrm{mmt}}) \in [3.68, 3.97]$. In Fig. 4, we see the relevant logarithmic plot of the correlation integral $C(\epsilon)$ over the neighborhood parameter ϵ.

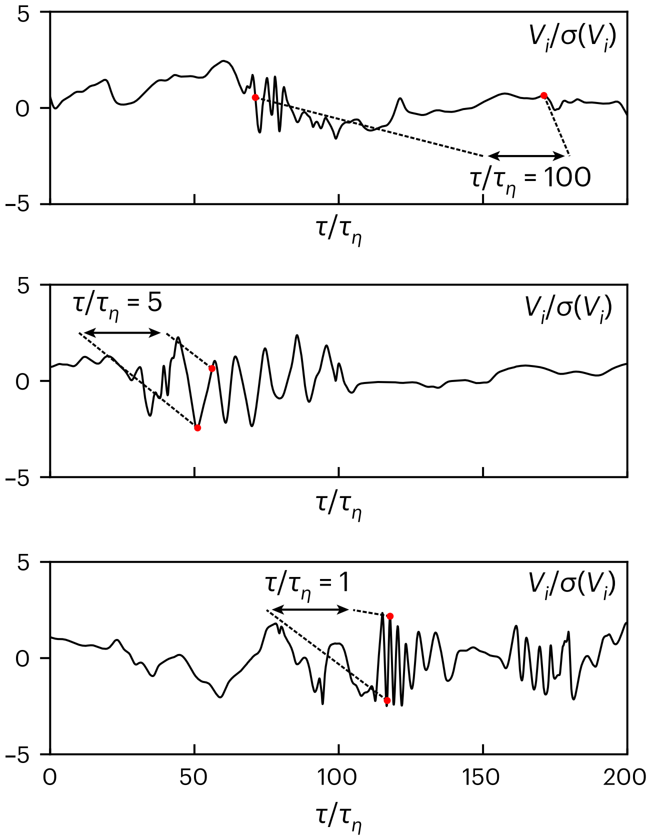

Fig. 5. This is Fig. 1(b,c,d) from [15], where, for three different trajectories, one component of the velocities V_i^k (denoted simply V_i in the figure), normalized by one standard deviation, is shown, over the $T \approx 200\tau_\eta$ length of the trajectory. From top to bottom, the sampling length of $\Delta_\tau V_i^k(t)$ is indicated for $\tau = 100\tau_\eta$, $5\tau_\eta$ or τ_η, respectively.

4.3. Lagrangian turbulence

The 3rd dataset that we consider consists of time-series of particle velocities $V(t)$, which are obtained by a high-resolution numerical simulation of the 3D Navier–Stokes equations with large-scale isotropic forcing [15,32]. Specifically, each particle i, follows the tracer dynamics

$$V_i(t) = u(X_i(t), t) \tag{22}$$

where u is a solution to the Navier–Stokes equations [33,34],

$$\begin{cases} \partial_t u + u \cdot \nabla u = -\nabla p + \nu \Delta u + F \\ \nabla \cdot u = 0 \end{cases} \tag{23}$$

and F is a homogeneous and isotropic forcing term which drives the flow to a non-equilibrium statistically steady state. Moreover, F is obtained via a 2nd order Ornstein–Uhlenbeck process [35,36], in which context 2 characteristic time scales are included; τ_E which is representative of the energy-containing scales of motion at 1st order and τ_η which is representative of the dissipative scales of motion, at 2nd order. For further details on the specifics of the numerical simulation the reader is referred to [37].

For the database used in this study and in [15], $N_p = 327680$ particles trajectories are tracked, each spanning a length of $T \approx 200\tau_\eta$, sampled every $dt \approx 0.1\tau_\eta$, so that each time series consists of $T/dt = 2000$, 3-dimensional velocities, $V_i(t_j) = (V_i^x(t_j), V_i^y(t_j), V_i^z(t_j)) \in \mathbb{R}^3$. These N_p particles are injected into the cubic-periodic domain of the simulation at random, once a statistically stationary evolution has been achieved for the underlying Eulerian flow. For the 1-dimensional diffusion model trained in [15], the three velocity components are not distinguished thus tripling the size of the available data to 983040 trajectories,

$$\mathcal{D}_{\text{full}} = \left\{ \left(V_i^k(t_1), \dots, V_i^k(t_{2000}) \right) \ : \ i \in \{1, \dots, 327680\} \text{ and } k \in \{x, y, z\} \right\} \tag{24}$$

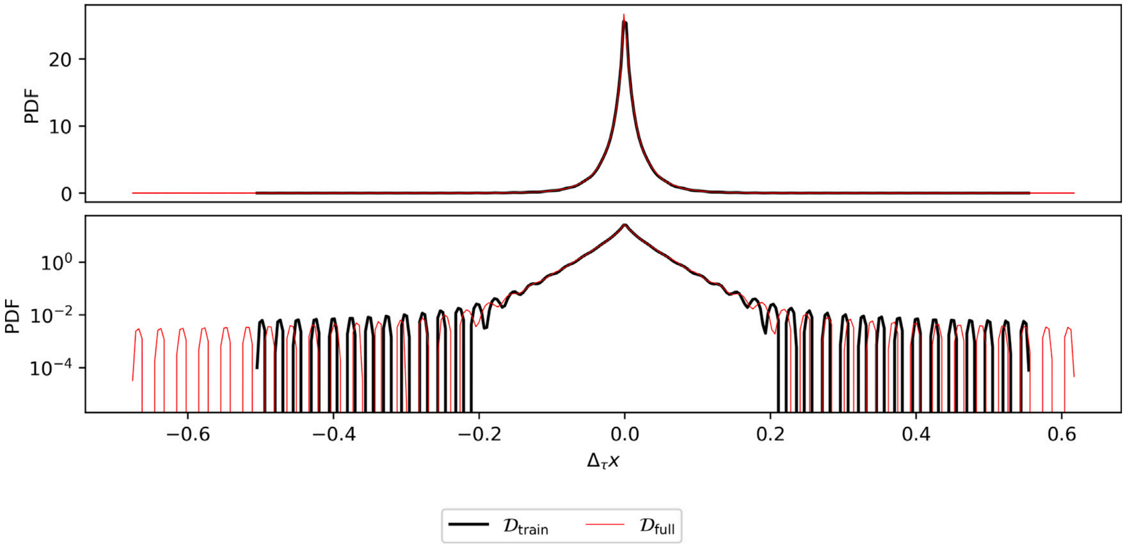

Fig. 6. PDFs of Δ_τ statistic (14), in linearly (top) and logarithmically (bottom) scaled axes, for the training and full on-hyperball dataset following the distribution defined in Eqs. (13).

Finally, all the observables of interest primarily rely on the statistics of the finite time differences $\Delta_\tau V_i^k(t) = V_i^k(t+\tau) - V_i^k(t)$, exactly as in Eqs. (14) and (20). The length of τ, relative to the entire trajectory length, for $\tau \in \{100\tau_\eta, 5\tau_\eta, \tau_\eta\}$ is visualized in Fig. 5, along with some indicative trajectories $V_i^k(t)$.

5. Results

To perform the numerical experiments, for each dataset $\mathcal{D}_{\text{full}}$, which is used for reference statistics, we retrieve another smaller dataset $\mathcal{D}_{\text{train}}$ from the same distribution to train the diffusion model. Specifically, for the on-hyperball dataset following the distribution defined through Eqs. (13), we generate approximately $330 \cdot 10^3$ points on a 2000-dimensional hyperball, which we denote as $\mathcal{D}_{\text{full}}$, and then we train a diffusion model on another dataset following the same distribution of only $15 \cdot 10^3$ points. The respective PDFs, of $\Delta_\tau x$ are shown in Fig. 6.

On the other hand, for the MMT dataset, we generate $57 \cdot 10^3$ trajectories, which we denote as $\mathcal{D}_{\text{full}}$, and then we generate another set of trajectories of size $3.7 \cdot 10^3$, on which we train a diffusion model. Details about the hyperparameters chosen in the training of both models can be found in Appendix D.

Before proceeding with the results from the diffusion models trained on these three datasets, we first apply criterion 3.1. For the on-hyperball dataset, it is evident by looking at Fig. 7, that for each point $\mathbf{x} = [x_1, \ldots, x_n]$, the quantities $\|\mathbf{x}\|$ and $\max_i |\Delta_\tau|x_i\; \|$ are positively correlated. In turn, this implies that, for this dataset, the extreme events as defined through the statistic $\Delta_\tau|x_i|$, lie close to the manifold boundary.

For the MMT it is necessary to apply Algorithm 1 to determine how close the extreme events are to the manifold boundary. Since the algorithm depends on the parameter δ, which determines the size of the neighborhood used to calculate the boundary degrees $\partial_\delta \mathbf{u}$ of each datapoint $\mathbf{u}$, the choice of δ needs to be investigated. We begin by defining as extreme events those points $\mathbf{u}$, for which $\max_i |\Delta_\tau|u_i\; \|$ is further than 2 standard deviations from the mean of the entire dataset. Then, for any one δ, we classify a point $\mathbf{u}$ as lying in the interior of $\mathcal{M}$ if $\partial_\delta \mathbf{u}$ is less than the mean of $\partial_\delta \mathbf{v}$ plus a single standard deviation over all $\mathbf{v}$ in the dataset (further details and discussion on this process are provided in Appendix C). We can then calculate, for each δ, the percentage of extreme events which lie in the interior of the manifold. In Fig. 8 we plot the boundary degrees 3.3 of the MMT dataset, for some indicative parameters $\delta \in [16, 19]$ (again for details on how this interval is selected we refer to Appendix C), parametrized according to $\max|\Delta_\tau u|$ exhibited in the relevant sample. The red dashed line is the cutoff, point below which, all points are classified as interior points. It becomes evident that for this case, the extreme events lie predominantly in the interior of the manifold.

Proceeding with the Lagrangian dataset used in [15], we repeat the previous process of applying Algorithm 1, again defining extreme events and boundary points in the same way as before. There are four statistics of the dataset that are investigated, each a finite difference $\Delta_\tau V_i$ of the velocity trajectories, for different values of τ. Specifically, for τ_η one of the characteristic time scales of the Lagrangian turbulence problem (see Section 4.3), the authors in [15] investigate the cases when $\tau/\tau_\eta \in \{1, 2, 5, 100\}$. Plotting the relevant statistic of the datapoints against their boundary degrees in Figs. 9–12, respectively for each τ/τ_η, we see that for all four cases, a significant proportion of the resulting extremes lie in the interior of the manifold.

Thus, Criterion 3.1 implies that the diffusion model will do much better on capturing the tails of the observables of interest in the MMT and Lagrangian datasets compared with the on-hyperball dataset.

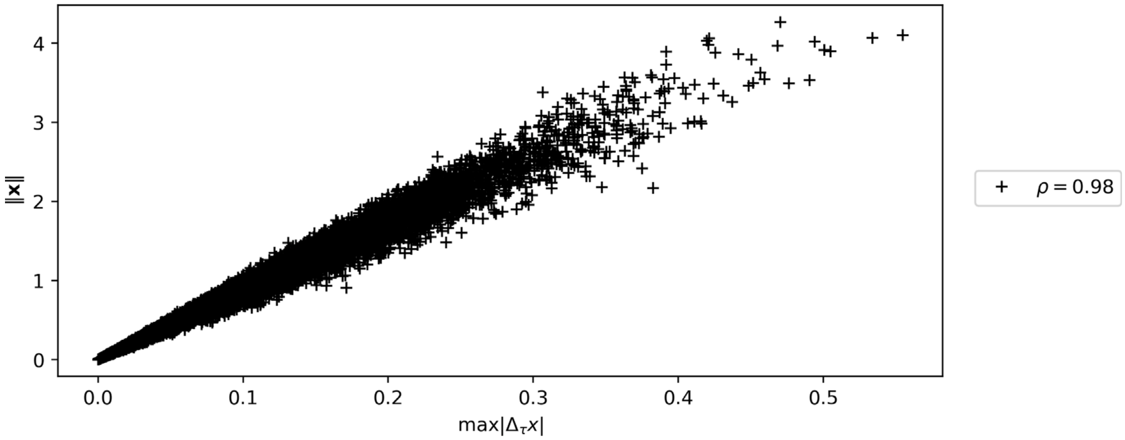

Fig. 7. Norm $\|\mathbf{x}\|$ of the elements $\mathbf{x} \in \mathcal{M}$ of the on-hyperball dataset (see Eq. (13)), for various parameters, parametrized according to $\max|\Delta_\tau u|$ exhibited in the relevant sample. In the legend, the respective correlation coefficient is reported. It is evident that there is a strong positive correlation between the distance of samples to the boundary of the underlying manifold $\mathcal{M}$ and the most extreme occurrences of the statistic Δ_τ, so that the relevant extreme events are expected to lie close to the boundary of $\mathcal{M}$.

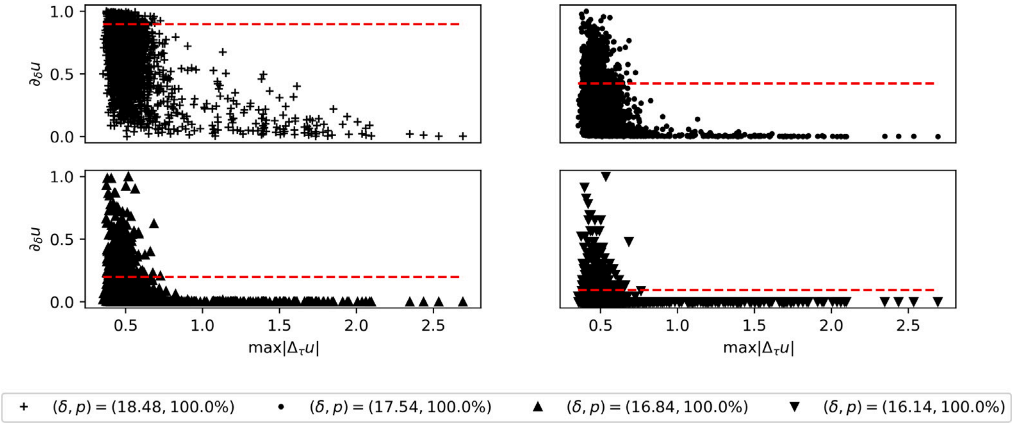

Fig. 8. Boundary degrees 3.3 of the MMT dataset (see Section 4.2.1), for various parameters $\delta \in [16, 19]$, parametrized according to $\max|\Delta_\tau u|$ exhibited in the relevant sample. In the legend, we report the δ parameter used in Algorithm 1 to calculate the relevant boundary degrees and the percentage, p, of extreme events (u such that $\max|\Delta_\tau u| > \mu + 2\sigma$) that are classified as interior points. It becomes evident that for this case, the extreme events lie predominantly in the interior of the manifold.

Indeed, observing Fig. 13 that involves the on-hyperball dataset, we note that the blue curve, which is based on $320 \cdot 10^3$ samples generated by the diffusion model, is unable to capture the tails past ±0.2. On the other hand, for the MMT dataset in Fig. 14, the shape of the tails generated by the trained diffusion model using $57 \cdot 10^3$ samples, is captured correctly from $[-0.7, 0.5]$ to ±1.5. It should be emphasized that while the tail is captured accurately, the absolute value of the probability of the tails is not. This may be associated with the sharp value of the PDF at 0, which requires a large number of data points to capture, not available in this experiment. Similarly, for the Lagrangian dataset, looking at Fig. 15, it is evident that the diffusion model trained on 10% of the available data (green line) is able to successfully extrapolate the statistics of its training dataset, for all $\tau/\tau_\eta = \{1, 2, 5, 100\}$.

6. Conclusions

We have formulated a generic geometric criterion to project the performance of diffusion models on capturing extreme statistics for events not necessarily present in the training data. The key idea behind the criterion is to evaluate whether the generative algorithm will need to operate in an interpolation or extrapolation mode. For the case where extreme events ‘live’ on the boundary of the dataset we expect that generative algorithms will have poor performance, simply because they do not have the underlying information to extrapolate statistics. On the other hand, if the extreme events are embedded in the manifold of data, it is expected that generative algorithms will have the capacity to generate a large number of samples that can eventually reproduce extreme event statistics. We have employed a geometric algorithm that is capable of computing the criterion even for very high-dimensional datasets.

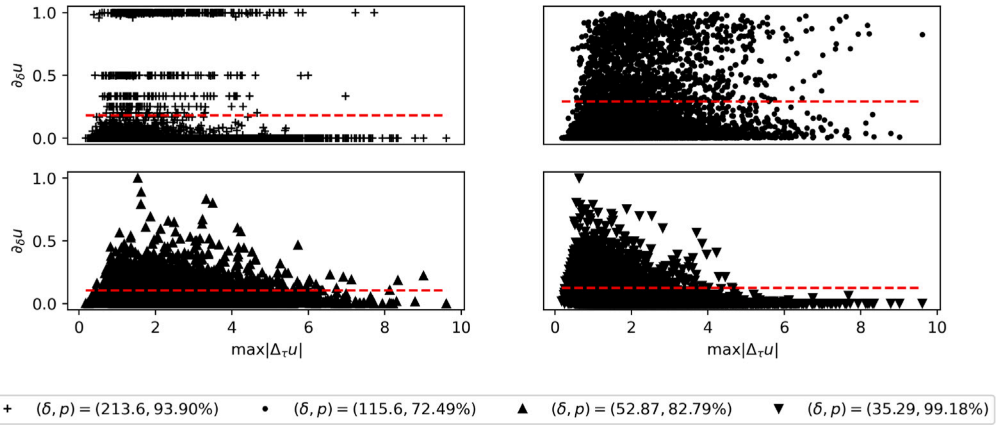

Fig. 9. Boundary degrees 3.3 of the Lagrangian Turbulence dataset (see [15]) and statistic corresponding to $\tau/\tau_\eta = 1$, for various parameters δ, parametrized according to $\max|\Delta_\tau u|$ exhibited in the relevant sample. In the legend, we report the δ parameter used in Algorithm 1 to calculate the relevant boundary degrees and the percentage, p, of extreme events (u such that $\max|\Delta_\tau u| > \mu + 2\sigma$) that are classified as interior points. It becomes evident that for this case, a high percentage of the extreme events lie in the interior of the manifold. Across all $\delta \in [25, 260]$, at least 70% of extreme events are classified as lying in the interior points.

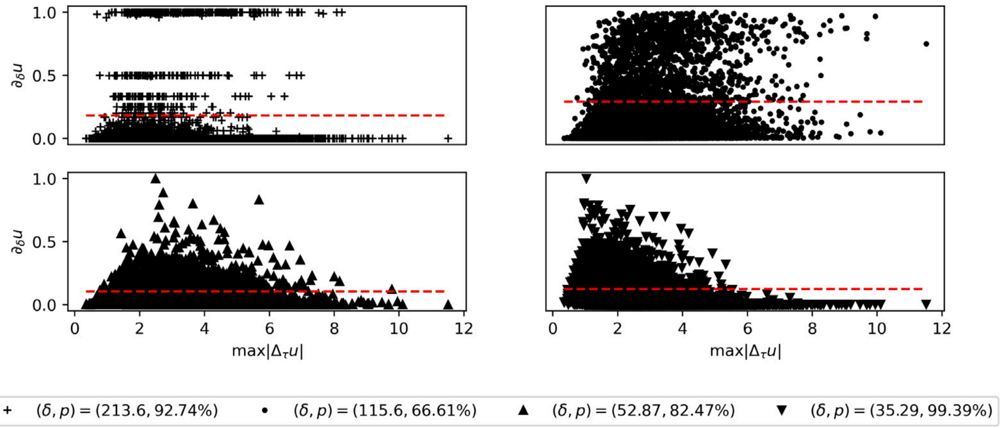

Fig. 10. Boundary degrees 3.3 of the Lagrangian Turbulence dataset (see [15]) and statistic corresponding to $\tau/\tau_\eta = 2$, for various parameters δ, parametrized according to $\max|\Delta_\tau u|$ exhibited in the relevant sample. In the legend, we report the δ parameter used in Algorithm 1 to calculate the relevant boundary degrees and the percentage, p, of extreme events (u such that $\max|\Delta_\tau u| > \mu + 2\sigma$) that are classified as interior points. It becomes evident that for this case, a high percentage of the extreme events lie in the interior of the manifold. Across all $\delta \in [25, 260]$, at least 65% of extreme events are classified as lying in the interior points.

We have illustrated the geometric criterion on three high-dimensional datasets with very different properties: in the first dataset the extreme events lie on the boundary and in the other two cases on the interior of the dataset. As expected, poor performance in capturing tail statistics is observed in the dataset where extremes are on the boundary of the dataset, while for other two, where extremes are in the interior, the tails predicted by the diffusion model compare favorably with the reference statistics. The developed criterion can be used as a guideline, an a priori test, to predict the performance in capturing extreme event statistics for problems where data is not plentiful. Further, since the criterion is not informed by the specific DDPM architecture, we expect it to be applicable to a larger class of generative models, such as GANs. It is then reasonable, for further research, to investigate the applicability/ non-applicability of the criterion on a wider domain than presented here.

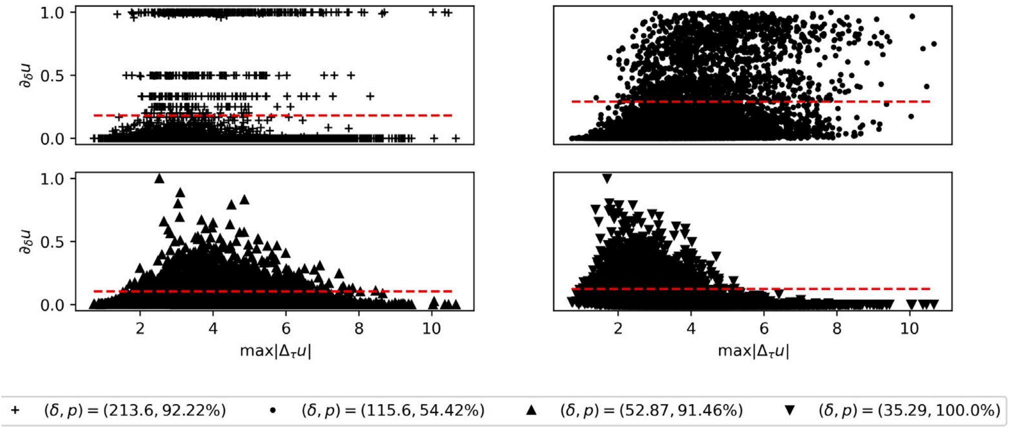

Fig. 11. Boundary degrees 3.3 of the Lagrangian Turbulence dataset (see [15]) and statistic corresponding to $\tau/\tau_\eta = 5$, for various parameters δ, parametrized according to $\max|\Delta_\tau u|$ exhibited in the relevant sample. In the legend, we report the δ parameter used in Algorithm 1 to calculate the relevant boundary degrees and the percentage, p, of extreme events (u such that $\max|\Delta_\tau u| > \mu + 2\sigma$) that are classified as interior points. It becomes evident that for this case, a high percentage of the extreme events lie in the interior of the manifold. Across all $\delta \in [25, 260]$, at least 54% of extreme events are classified as lying in the interior points.

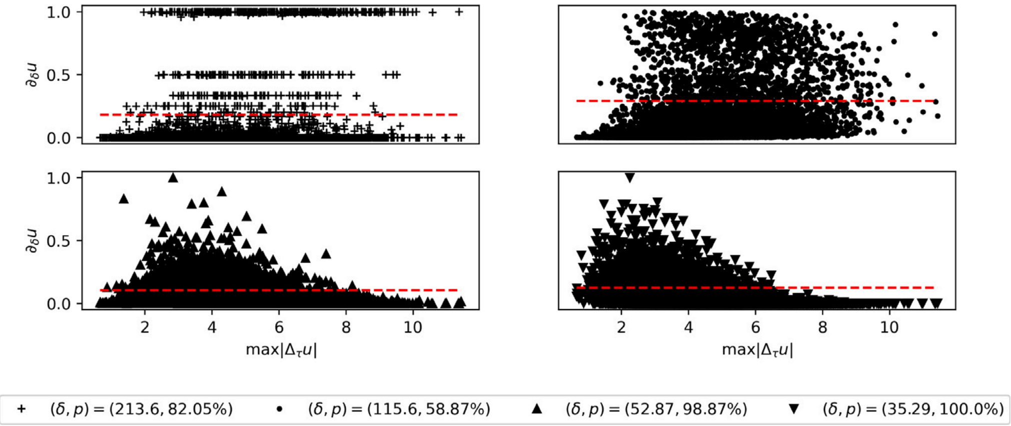

Fig. 12. Boundary degrees 3.3 of the Lagrangian Turbulence dataset (see [15]) and statistic corresponding to $\tau/\tau_\eta = 100$, for various parameters δ, parametrized according to $\max|\Delta_\tau u|$ exhibited in the relevant sample. In the legend, we report the δ parameter used in Algorithm 1 to calculate the relevant boundary degrees and the percentage, p, of extreme events (u such that $\max|\Delta_\tau u| > \mu + 2\sigma$) that are classified as interior points. It becomes evident that for this case, a high percentage of the extreme events lie in the interior of the manifold. Across all $\delta \in [25, 260]$, at least 46% of extreme events are classified as lying in the interior points.

CRediT authorship contribution statement

Stamatis Stamatelopoulos: Writing – original draft, Methodology, Investigation, Formal analysis, Data curation, Conceptualization. **Themistoklis P. Sapsis:** Writing – original draft, Supervision, Methodology, Investigation, Funding acquisition, Formal analysis, Conceptualization.

Funding

This work has been supported by the AFOSR, United States of America grant FA9550-23-1-0517 and ONR, United States of America grant N00014-21-1-2357.

Declaration of competing interest

The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: Themistoklis Sapsis reports financial support was provided by Massachusetts Institute of Technology. If there are other authors, they declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

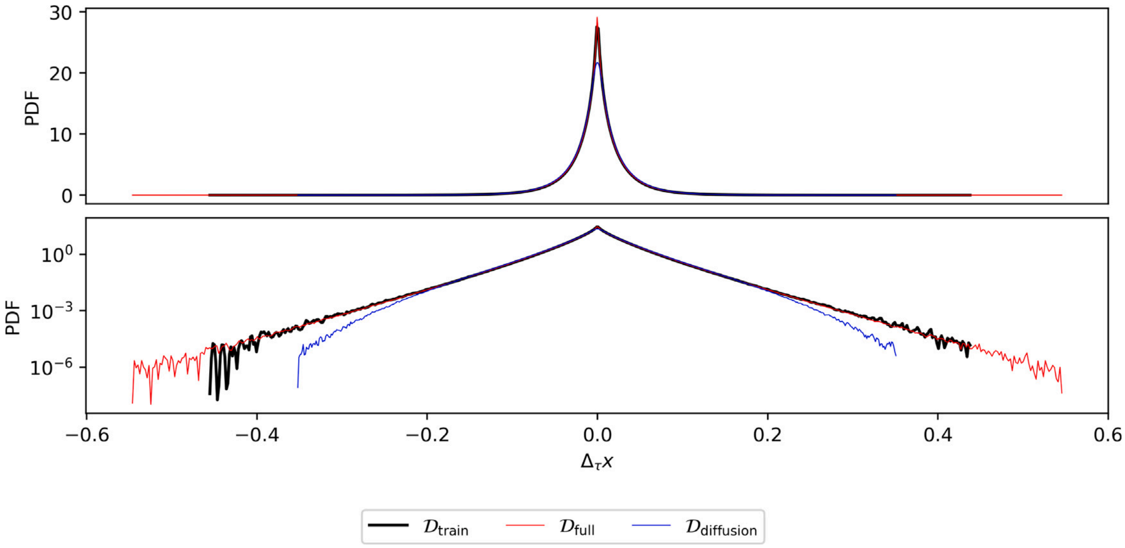

Fig. 13. PDFs of Δ_τ statistic (14), in linearly (top) and logarithmically (bottom) scaled axes, for the on-hyperball dataset following the distribution defined in Eqs. (13). The blue-colored PDF represents a dataset generated through a diffusion model trained on the training dataset $\mathcal{D}_{\text{train}}$, while $\mathcal{D}_{\text{full}}$ is a larger dataset following the same distribution as $\mathcal{D}_{\text{train}}$. In particular, $|\mathcal{D}_{\text{train}}| = 15 \cdot 10^3$, $|\mathcal{D}_{\text{diffusion}}| = 320 \cdot 10^3$ and $|\mathcal{D}_{\text{full}}| = 330 \cdot 10^3$.

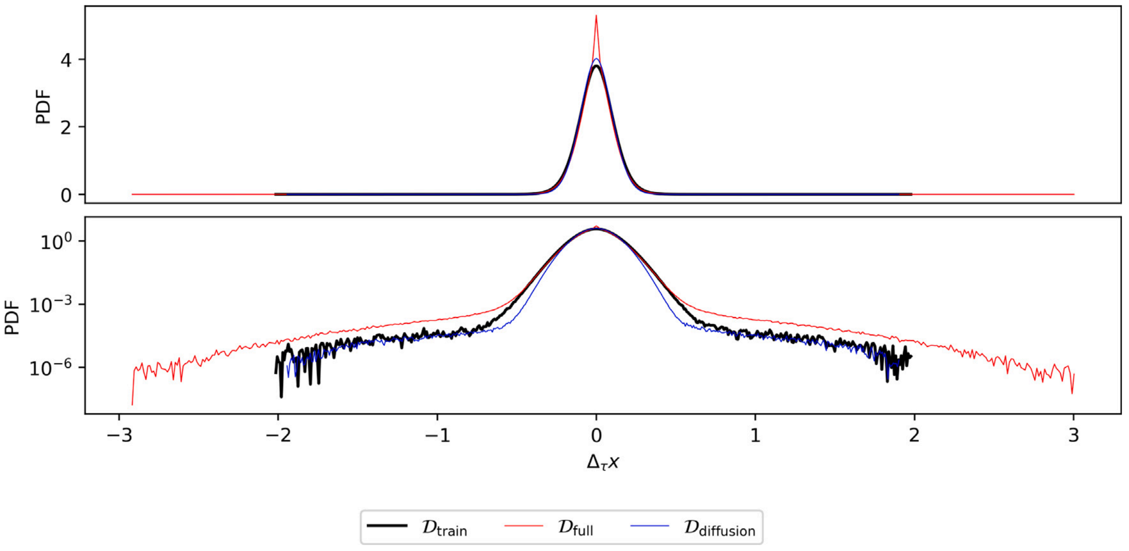

Fig. 14. PDFs of Δ_τ statistic (20), in linearly (top) and logarithmically (bottom) scaled axes, for the dataset following the distribution defined in Section 4.2.1. The blue-colored PDF represents a dataset generated through a diffusion model trained on the training dataset $\mathcal{D}_{\text{train}}$, while $\mathcal{D}_{\text{full}}$ is a larger dataset following the same distribution as $\mathcal{D}_{\text{train}}$. In particular, $|\mathcal{D}_{\text{train}}| = 3.7 \cdot 10^3$, $|\mathcal{D}_{\text{diffusion}}| = 120 \cdot 10^3$ and $|\mathcal{D}_{\text{full}}| = 57 \cdot 10^3$.

Appendix A. Elementary geometry of high dimensional balls

A n-dimensional ball of radius r centered at $\mathbf{c}$ is defined as

$$\mathbf{B}^n(\mathbf{c}; r) = \{\mathbf{x} \in \mathbb{R}^n : \ \|\mathbf{c} - \mathbf{x}\| \leq r\} \tag{25}$$

where if $\mathbf{c}$ is omitted, it will be taken to equal $\mathbf{0}$ where its n-dimensional volume is given by

$$V(\mathbf{B}^n(\mathbf{c}; r)) = \frac{\pi^{n/2} r^n}{\Gamma(n/2 + 1)} \tag{26}$$

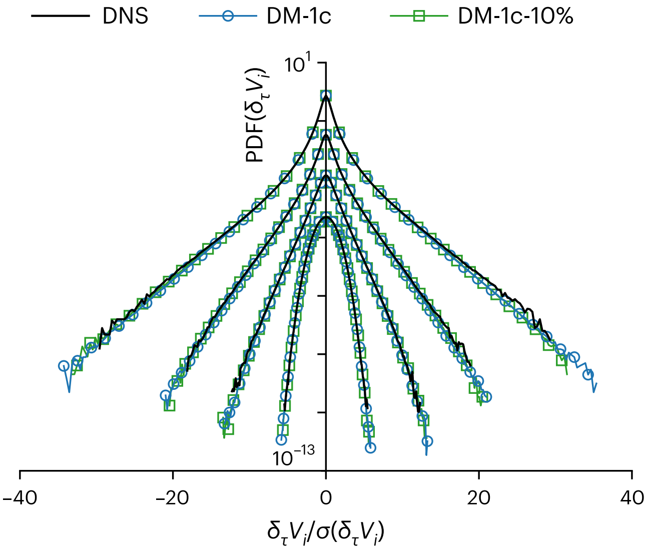

Fig. 15. This is Figure 1a from [15], where the PDFs for finite increments of the velocities V_i are shown for $\tau/\tau_\eta = 1, 2, 5, 100$, from bottom to top. The PDFs for different τ are shifted for the sake of presentation. The black lines correspond to the dataset generated through Direct Numerical Simulation (DNS), which corresponds to $\mathcal{D}_{\text{full}}$ in our notation. The blue lines correspond to the Diffusion Model on 1-Component (DM-1c) that the authors trained on $\mathcal{D}_{\text{full}}$. The green lines (DM-1c-10%) correspond to the diffusion model that the authors trained on 10% of $\mathcal{D}_{\text{full}}$, which is $\mathcal{D}_{\text{diffusion}}$ in our notation. It is evident that the diffusion model is able to successfully extrapolate the statistics of its training dataset.

Now, consider the ball $\mathbf{B}^n(\mathbf{c}; 0.5)$, $\mathbf{c} = (0.5, \ldots, 0.5)^T$ inscribed in the unit hypercube $[0, 1]^n$. It is easy to see that the volume occupied by $\mathbf{B}^n$ is dominated by that $[0, 1]^n$ as the dimension increases,

$$\frac{V(\mathbf{B}^n(\mathbf{c}; 0.5))}{V([0, 1]^n)} = \frac{(\sqrt{\pi}/2)^n}{\Gamma(n/2 + 1)} \to 0, \qquad n \to \infty$$

Next, define the n-dimensional shell of $r_1 < r_2$, centered at $\mathbf{c}$ as,

$$\mathbf{Sh}^n(\mathbf{c}; r_1; r_2) = \mathbf{B}^n(\mathbf{c}; r_2)/\mathbf{B}^n(\mathbf{c}; r_1) = \{\mathbf{x} \in \mathbb{R}^n : \ r_1 < \|\mathbf{x} - \mathbf{c}\| \le r_2\} \tag{27}$$

Based on (26) it is trivial to write,

$$V(\mathbf{Sh}^n(\mathbf{c}; r_1; r_2)) = \frac{\pi^{n/2} r^n}{\Gamma(n/2 + 1)}(r_2^n - r_1^n) \tag{28}$$

Now, it is interesting to compare the volumes of $\mathbf{B}^n(R)$ and $\mathbf{B}^n(r)$ for $r \in [0, R]$. Specifically,

$$\frac{V(\mathbf{B}^n(r))}{V(\mathbf{B}^n(R))} = \left(\frac{r}{R}\right)^n \to 0, \qquad n \to \infty \tag{29}$$

which means that in high enough dimensions, the volume of $\mathbf{B}^n(R)$ is concentrated in $\mathbf{Sh}^n(\mathbf{c}; R - \delta r; R)$.

Appendix B. Solution scheme for the MMT equations

In this section we provide details on how we solved Eq. (18), by employing ETD4RK [28] scheme. Specifically, given an ordinary differential equation of the form,

$$\dot{\psi}(t) = c\psi(t) + F(\psi(t), t) \tag{30}$$

with initial condition $\psi(t_0) = \psi_0$, let

$$a_n = \psi_n e^{ch/2} + (e^{ch/2} - 1)F(\psi_n, t_n)/c, \tag{31a}$$

$$b_n = \psi_n e^{ch/2} + (e^{ch/2} - 1)F(a_n, t_n + h/2)/c, \tag{31b}$$

$$c_n = a_n e^{ch/2} + (e^{ch/2} - 1)(2F(b_n, t_n + h/2) - F(\psi_n, t_n))/c, \tag{31c}$$

$$\begin{aligned} \psi_{n+1} = \psi_n e^{ch} + \Big(& F(\psi_n, t_n)(-4 - hc - e^{ch}(4 - 3hc + h^2c^2)) \\ & + 2(F(a_n, t_n + h/2) + F(b_n, t_n + h/2))(2 + hc + e^{ch}(-2 + hc)) \\ & + F(c_n, t_n + h)(-4 - 3hc - h^2c^2 + e^{ch}(4 - hc)) \Big) / (h^2c^3) \end{aligned} \tag{31d}$$

where $h = t_{n+1} - t_n$. To utilize (30), we move (18) into Fourier space by multiplying with e^{-ixk} and integrating over x,

$$i\widehat{\partial_t u}(t;k) = \widehat{|\partial_x|^{1/2}}u(t;k) + \lambda\widehat{|u|^2}u(t;k) + i\widehat{D}u(t;k)$$

so that if we factor ∂_t out of the first term and expand the second and fourth terms according to (16) and (17) respectively,

$$\partial_t\hat{u}(t;k) = \begin{cases} ((|k|-k^*)^2 - i\sqrt{|k|})\hat{u}(t;k) - i\lambda\widehat{|u|^2 u}(t;k) & |k| > k^* \\ -i\sqrt{|k|}\hat{u}(t;k) - i\lambda\widehat{|u|^2 u}(t;k) & |k| \leq k^* \end{cases} \tag{32}$$

which for any choice of k is clearly in the form (30). We should note that in order to solve (32) using (31a), it is necessary to evaluate the so-called ϕ-functions,

$$\phi_1(z) = \frac{e^z - 1}{z} \quad \text{and} \quad \phi_3(z) = \frac{e^z - z^2/2 - z - 1}{z^3}$$

which suffer from numerical cancellation errors for $|z| \to 0$ [38]. To circumvent this, we utilize the EXPRINT library [39] which employs Padé approximation for the relevant computations. To proceed, we retrieve the initial conditions $u(0,x)$ from Eq. (19) and transport them to Fourier space, symmetrizing them by removing their highest mode, and set $\psi_0(k) = \widehat{u(0,x)}(k)$.

Appendix C. Selecting the δ parameter for BRIM

As mentioned in Section 5, for the MMT and the Lagrangian datasets, it is necessary to apply Algorithm 1 in order to determine how close the extreme events are to the manifold boundary. In this appendix, we go into detail on the process of choosing a range for the parameter δ when applying Algorithm 1.

This parameter determines the size of the neighborhood used to calculate the boundary degrees, $\partial_\delta \mathbf{u}$, of each datapoint $\mathbf{u}$. In general, there is a range $[\delta_{\min}, \delta_{\max}]$, with $\partial_\delta \mathbf{u} = \partial_{\delta_{\min}} \mathbf{u}$ for all $\delta \leq \delta_{\min}$ and $\partial_\delta \mathbf{u} = \partial_{\delta_{\max}} \mathbf{u}$ for all $\delta \geq \delta_{\max}$, so that we need to only investigate $\delta \in [\delta_{\min}, \delta_{\max}]$. Next, recall that, as explained in Section 3, not all of the values within $[\delta_{\min}, \delta_{\max}]$ are admissible, since for large values of δ, local features of the manifold are obscured. On the other hand, for small values, the normal vector at the manifold boundary is erroneously approximated. Further, denoting $m(\delta) = \min_{\mathbf{u}} \partial_\delta \mathbf{u}$ and $M(\delta) = \max_{\mathbf{u}} \partial_\delta \mathbf{u}$, we note that for large datasets, we expect that when δ is admissible, intermediate values in $[m(\delta), M(\delta)]$ will also be attained by $\partial_\delta \mathbf{u}$. One way to see this, is to first consider the limiting process of increasing the size of the dataset, in such a way that the continuous manifold $\mathcal{M}$ is recovered, with approximately uniform density in its interior across all steps of the process. Then, $\partial_\delta \mathbf{u}(t)$ is expected to be continuous on any on-manifold trajectory $u(t) : \mathbb{R} \to \mathcal{M}$, which implies that for discrete but reasonably large datasets, if $\partial_\delta \mathbf{u}$ is strongly binary (i.e. all values are either very close to $m(\delta)$ and/or $M(\delta)$), then this is an indicator of inadmissibility for the specific δ. For instance, for the MMT dataset we get $[\delta_{\min}, \delta_{\max}] \approx [16, 23]$, but for $\delta \in [20, 23]$ this desirable non-binary behavior just described is not apparent (see Fig. 16), rendering the admissible range of δ as a subset of $[16, 20]$.

We can now begin the investigation of the positions of the extreme events of the MMT dataset for all δ. To do this, we define as extreme events those points $\mathbf{u}$, for which $\max_i |\Delta_\tau |u_i\ \||$ is further than 2 standard deviations from the mean of the entire dataset. Then, for any one δ, we classify a point $\mathbf{u}$ as lying in the interior of $\mathcal{M}$ if $\partial_\delta \mathbf{u}$ is less than the mean of $\partial_\delta \mathbf{v}$ plus a single standard deviation over all $\mathbf{v}$ in the dataset. The addition of a standard deviation is added to take into account the variability of the boundary degrees of interior points from the fact that the points are not evenly distributed in the ambient space. We can then calculate, for each δ, the percentage of extreme events which lie in the interior of the manifold. Looking at Fig. 17, it is evident that, indeed, a very small portion of the extreme events on the MMT data manifold lie close to its boundary.

Proceeding with the Lagrangian dataset used in [15], we repeat the previous calculation, again defining extreme events and boundary points in the same way as before. There are 4 statistics of the dataset that are investigated, each a finite difference $\Delta_\tau V_i$ of the velocity trajectories, for different values of τ. Specifically, for τ_η one of the characteristic time scales of the Lagrangian turbulence problem in [15], the authors investigate the cases when $\tau/\tau_\eta \in \{1, 2, 5, 100\}$. The resulting range is now $\delta \in [25, 260]$, and looking at Fig. 18, we see that for all four cases of τ/τ_η, a significant proportion of the resulting extremes lie in the interior of the manifold. Since this is the case for all values of δ, no further investigation into which δ are admissible is necessary.

Appendix D. Diffusion model hyperparameters

In this section we document the exact hyperparameters used to train the diffusion models used in this study. The diffusion model framework we employ is identical to that employed in [15] so that for further details on the specific hyperparameters the reader is referred there. For the on-hyperball dataset (see Fig. 13), we have the following list of hyperparameters

- diffusion_steps= 3000
- noise_schedule= tanh 2, 0.33
- num_channels= 128
- num_res_blocks= 3
- channel_mult= 1, 1, 2, 3, 4
- attention_resolutions= 250, 125
- batch_size= 32
- learning_rate= 10^{-4}

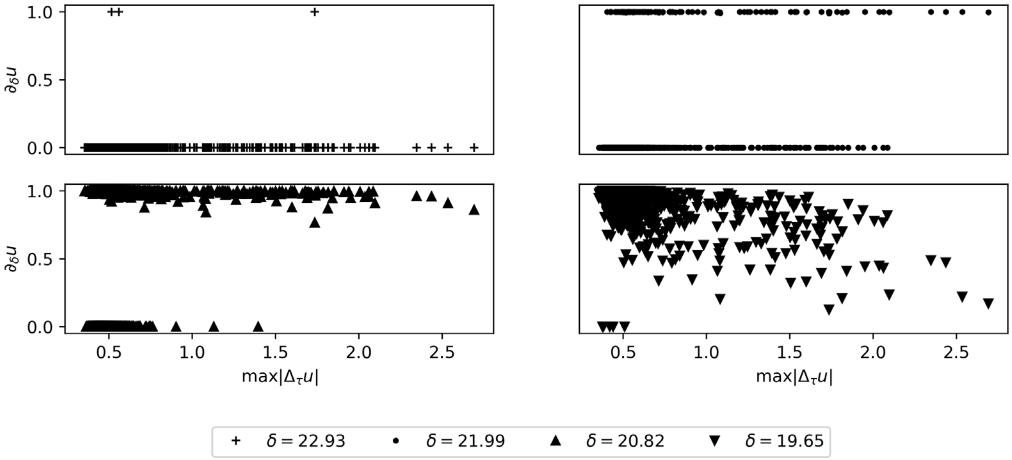

Fig. 16. Boundary degrees 3.3 of the MMT dataset (see Section 4.2.1), for $\delta \in \{19.65, 20.82, 21.99, 22.93\}$, parametrized according to $\max|\Delta_\tau u|$ exhibited in the relevant sample. It is evident that the pseudo-continuity property expected from admissible values of δ is not exhibited for $\delta < 20$, rendering the range $\delta \in [20, 23]$ inadmissible.

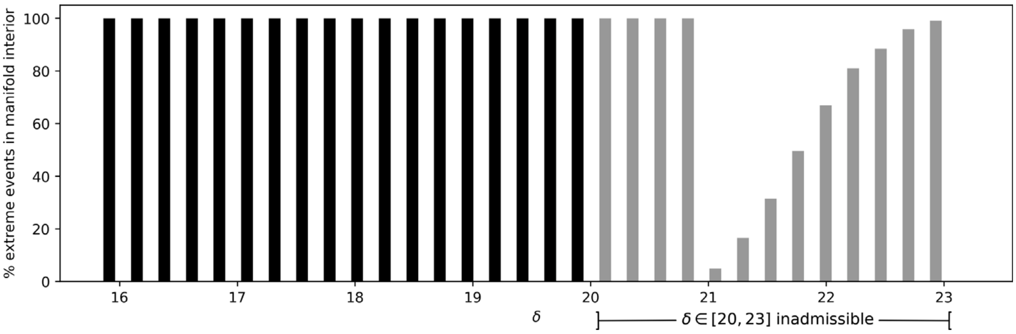

Fig. 17. Percentage of extreme events in the MMT dataset (see Section 4.2.1), which lie in the interior of the manifold, as determined by Algorithm 1 for all $\delta \in [\delta_{\min}, \delta_{\max}]$. Extreme events are characterized as those further than 2 standard deviations away from the mean. It is evident that for all admissible values of δ, there is a significant proportion of extreme events in the interior of the MMT data manifold.

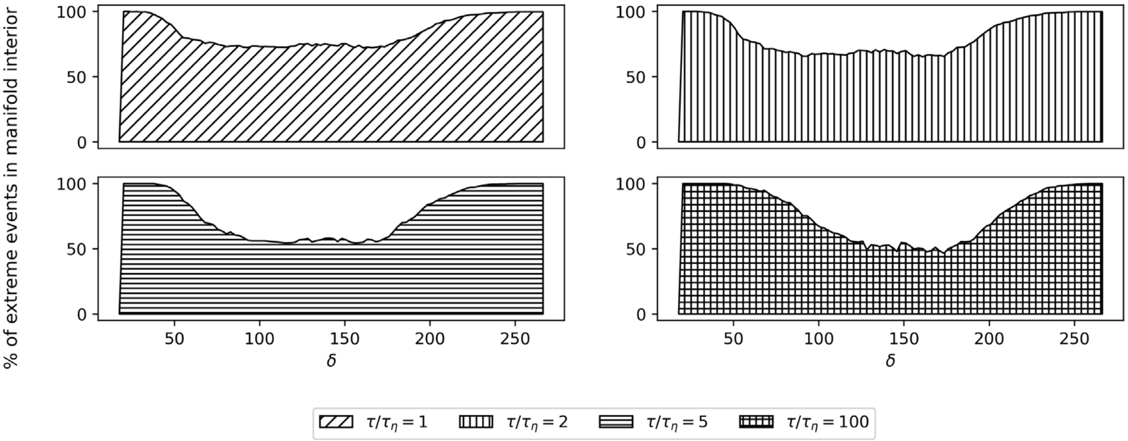

Fig. 18. Percentage of extreme events in the Lagrangian Turbulence dataset (see [15]), which lie in the interior of the manifold, as determined by Algorithm 1 for all $\delta \in [\delta_{\min}, \delta_{\max}]$. Extreme events are characterized as those further than 2 standard deviations away from the mean. It is evident that for all admissible values of δ, there is a significant proportion of extreme events in the interior of the Lagrangian Turbulence data manifold.

while for the MMT dataset (see Fig. 14), we get,

- `diffusion_steps`= 800
- `noise_schedule`= tanh 6, 1
- `num_channels`= 128
- `num_res_blocks`= 3
- `channel_mult`= 1, 1, 2, 3, 4
- `attention_resolutions`= 250, 125
- `batch_size`= 32
- `learning_rate`= 10^{-4}

Finally, for the Lagrangian dataset (see Fig. 15), the hyperparameters that the authors in [15] used are,

- `diffusion_steps`= 800
- `noise_schedule`= tanh 6, 1
- `num_channels`= 128
- `num_res_blocks`= 3
- `channel_mult`= 1, 1, 2, 3, 4
- `attention_resolutions`= 250, 125
- `batch_size`= 256
- `learning_rate`= 10^{-4}

Data availability

Data will be made available on request.

References

[1] P. Müller, C. Garrett, A. Osborne, Rogue waves—The fourteenth 'aha huliko'a hawaiian winter workshop, Oceanography (2005) http://dx.doi.org/10.5670/oceanog.2005.30.

[2] K. Dysthe, H.E. Krogstad, P. Müller, Oceanic rogue waves, Annu. Rev. Fluid Mech. 40 (Volume 40, 2008) (2008) 287–310, http://dx.doi.org/10.1146/annurev.fluid.40.111406.102203, URL https://www.annualreviews.org/content/journals/10.1146/annurev.fluid.40.111406.102203.

[3] C. Kharif, E. Pelinovsky, A. Slunyaev, Observation of rogue waves, in: Rogue Waves in the Ocean, Springer Berlin Heidelberg, Berlin, Heidelberg, 2009, pp. 11–31, http://dx.doi.org/10.1007/978-3-540-88419-4_2.

[4] M. Onorato, S. Residori, U. Bortolozzo, A. Montina, F. Arecchi, Rogue waves and their generating mechanisms in different physical contexts, Phys. Rep. 528 (2) (2013) 47–89, http://dx.doi.org/10.1016/j.physrep.2013.03.001, Rogue waves and their generating mechanisms in different physical contexts, URL https://www.sciencedirect.com/science/article/pii/S0370157313000963.

[5] A. Toffoli, Rogue waves in random sea states: An experimental perspective, in: M. Onorato, S. Resitori, F. Baronio (Eds.), Rogue and Shock Waves in Nonlinear Dispersive Media, Springer International Publishing, Cham, 2016, pp. 179–203, http://dx.doi.org/10.1007/978-3-319-39214-1_7.

[6] S.K. Haver, A possible freak wave event measured at the draupner jacket january 1 1995, 2004, URL https://api.semanticscholar.org/CorpusID:131798586.

[7] P. Liu, A chronology of freauqe wave encounters, Geofizika (geofizika-journal@gfz.hr) 24 (1) (2007).

[8] W. Cousins, M. Onorato, A. Chabchoub, T.P. Sapsis, Predicting ocean rogue waves from point measurements: An experimental study for unidirectional waves, Phys. Rev. E 99 (2019) 032201, http://dx.doi.org/10.1103/PhysRevE.99.032201, URL https://link.aps.org/doi/10.1103/PhysRevE.99.032201.

[9] C. Genest, A.-C. Favre, Everything you always wanted to know about copula modeling but were afraid to ask, J. Hydrol. Eng. 12 (4) (2007) 347–368, http://dx.doi.org/10.1061/(ASCE)1084-0699(2007)12:4(347), URL https://ascelibrary.org/doi/abs/10.1061/%28ASCE%291084-0699%282007%2912%3A4%28347%29.

[10] P.K. Trivedi, D.M. Zimmer, Copula modeling: An introduction for practitioners, Found. Trends Econometr. 1 (1) (2007) 1–111, http://dx.doi.org/10.1561/0800000005, URL http://dx.doi.org/10.1561/0800000005.

[11] R.B. Nelsen, Definitions and basic properties, in: An Introduction To Copulas, Springer New York, New York, NY, 2006, pp. 7–49, http://dx.doi.org/10.1007/0-387-28678-0_2.

[12] M. Sklar, Fonctions de répartition à N dimensions et leurs marges, Annales de l'ISUP VIII (3) (1959) 229–231, URL https://hal.science/hal-04094463.

[13] J.E. Heffernan, J.A. Tawn, A conditional approach for multivariate extreme values (with discussion), J. R. Stat. Soc. Ser. B Stat. Methodol. 66 (3) (2004) 497–546, http://dx.doi.org/10.1111/j.1467-9868.2004.02050.x, arXiv:https://rss.onlinelibrary.wiley.com/doi/pdf/10.1111/j.1467-9868.2004.02050.x, URL https://rss.onlinelibrary.wiley.com/doi/abs/10.1111/j.1467-9868.2004.02050.x.

[14] A. Peard, J. Hall, Combining deep generative models with extreme value theory for synthetic hazard simulation: a multivariate and spatially coherent approach, 2023, arXiv:2311.18521, URL https://arxiv.org/abs/2311.18521.

[15] T. Li, L. Biferale, F. Bonaccorso, M.A. Scarpolini, M. Buzzicotti, Synthetic Lagrangian turbulence by generative diffusion models, Nat. Mach. Intell. 6 (4) (2024) 393–403, http://dx.doi.org/10.1038/s42256-024-00810-0.

[16] X. Zhong, L. Chen, J. Liu, C. Lin, Y. Qi, H. Li, Fuxi-extreme: Improving extreme rainfall and wind forecasts with diffusion model, 2023, arXiv:2310.19822, URL https://arxiv.org/abs/2310.19822.

[17] N.A. Letizia, A.M. Tonello, Segmented generative networks: Data generation in the uniform probability space, IEEE Trans. Neural Netw. Learn. Syst. 33 (3) (2022) 1338–1347, http://dx.doi.org/10.1109/TNNLS.2020.3042380.

[18] Y. Boulaguiem, J. Zscheischler, E. Vignotto, K. van der Wiel, S. Engelke, Modelling and simulating spatial extremes by combining extreme value theory with generative adversarial networks, 2022, arXiv:2111.00267, URL https://arxiv.org/abs/2111.00267.

[19] S. Bhatia, A. Jain, B. Hooi, ExGAN: Adversarial generation of extreme samples, 2021, arXiv:2009.08454, URL https://arxiv.org/abs/2009.08454.

[20] Association for Artificial Intelligence 2023, L. Chen, F. Du, Y. Hu, F. Wang, Z. Wang, SwinRDM: Integrate swinrnn with diffusion model towards high-resolution and high-quality weather forecasting, 2023, http://dx.doi.org/10.48448/ZN7F-FC64, URL https://underline.io/lecture/69034-swinrdm-integrate-swinrnn-with-diffusion-model-towards-high-resolution-and-high-quality-weather-forecasting.

[21] A. Nichol, P. Dhariwal, Improved denoising diffusion probabilistic models, 2021, arXiv:2102.09672, URL https://arxiv.org/abs/2102.09672.

[22] J. Sohl-Dickstein, E.A. Weiss, N. Maheswaranathan, S. Ganguli, Deep unsupervised learning using nonequilibrium thermodynamics, 2015, arXiv:1503.03585, URL https://arxiv.org/abs/1503.03585.

[23] J. Ho, A. Jain, P. Abbeel, Denoising diffusion probabilistic models, 2020, arXiv:2006.11239, URL https://arxiv.org/abs/2006.11239.

[24] B.-Z. Qiu, F. Yue, J.-Y. Shen, BRIM: An efficient boundary points detecting algorithm, in: Z.-H. Zhou, H. Li, Q. Yang (Eds.), Advances in Knowledge Discovery and Data Mining, Springer Berlin Heidelberg, Berlin, Heidelberg, 2007, pp. 761–768.

[25] C. Xia, W. Hsu, M. Lee, B. Ooi, BORDER: efficient computation of boundary points, IEEE Trans. Knowl. Data Eng. 18 (3) (2006) 289–303, http://dx.doi.org/10.1109/TKDE.2006.38.

[26] A. Majda, D. McLaughlin, T. E.G., A one-dimensional model for dispersive wave turbulence, Nonlinear Sci. (1997).

[27] W. Cousins, T.P. Sapsis, Quantification and prediction of extreme events in a one-dimensional nonlinear dispersive wave model, Physica D 280–281 (2014) 48–58, http://dx.doi.org/10.1016/j.physd.2014.04.012, URL https://www.sciencedirect.com/science/article/pii/S016727891400092X.

[28] S. Cox, P. Matthews, Exponential time differencing for stiff systems, Comput. Phys. (2002).

[29] O. Michel, P. Flandrin, Higher order statistics for chaotic signal analysis, in: C.T. Leondes (Ed.), Computer Techniques and Algorithms in Digital Signal Processing, in: Control and Dynamic Systems, Vol. 75, Academic Press, 1996, pp. 105–154, http://dx.doi.org/10.1016/S0090-5267(96)80040-5, URL https://www.sciencedirect.com/science/article/pii/S0090526796800405.

[30] P. Grassberger, I. Procaccia, Measuring the strangeness of strange attractors, Physica D 9 (1) (1983) 189–208, http://dx.doi.org/10.1016/0167-2789(83)90298-1, URL https://www.sciencedirect.com/science/article/pii/0167278983902981.

[31] G. Datseris, I. Kottlarz, A.P. Braun, U. Parlitz, Estimating fractal dimensions: A comparative review and open source implementations, Chaos 33 (10) (2023) http://dx.doi.org/10.1063/5.0160394.

[32] C. Calascibetta, L. Biferale, F. Borra, A. Celani, M. Cencini, Optimal tracking strategies in a turbulent flow, Commun. Phys. 6 (1) (2023) 256, http://dx.doi.org/10.1038/s42005-023-01366-y.

[33] U. Frisch, Turbulence: The legacy of A.N. Kolmogorov, Astrophys. Lett. Commun. (1995) http://dx.doi.org/10.1017/CBO9781139170666.

[34] S.B. Pope, Turbulent Flows, Cambridge University Press, 2000.

[35] S.B. Pope, Simple models of turbulent flowsa), Phys. Fluids 23 (1) (2011) 011301, http://dx.doi.org/10.1063/1.3531744, arXiv:https://pubs.aip.org/aip/pof/article-pdf/doi/10.1063/1.3531744/15723277/011301_1_online.pdf.

[36] B.L. Sawford, Reynolds number effects in Lagrangian stochastic models of turbulent dispersion, Phys. Fluids A: Fluid Dyn. 3 (6) (1991) 1577–1586, http://dx.doi.org/10.1063/1.857937, arXiv:https://pubs.aip.org/aip/pof/article-pdf/3/6/1577/12274485/1577_1_online.pdf.

[37] L. Biferale, F. Bonaccorso, M. Buzzicotti, C. Calascibetta, TURB-Lagr. A database of 3d Lagrangian trajectories in homogeneous and isotropic turbulence, 2024, arXiv:2303.08662, URL https://arxiv.org/abs/2303.08662.

[38] A. Kassam, L. Trefethen, Fourth-order time-stepping for stiff PDEs, Soc. Ind. Appl. Math. (2005).

[39] H. Berland, B. Skaflestad, W. Wright, EXPRINT-a MATLAB package for exponential integrators, ACM Trans. Math. Softw. (2007).