# Preprints.org

# Dual Theory of Decaying Turbulence. 2. Numerical Simulation and Continuum Limit

Alexander Migdal [*]

*Article*

# Dual Theory of Decaying Turbulence.
# 2. Numerical Simulation and Continuum Limit

**Alexander Migdal** ⬤

Department of Physics, New York University Abu Dhabi, Saadiyat Island, Abu Dhabi, PO Box 129188, Abu Dhabi, United Arab Emirates; am10485@nyu.edu

**Abstract:** This is the second paper in a cycle investigating the exact solution of loop equations in decaying turbulence. We perform numerical simulations of the Euler ensemble, suggested in the previous work, as a solution to the loop equations. We designed novel algorithms for simulation, which take a small amount of computer RAM so that only the CPU time grows linearly with the size of the system and its statistics. This algorithm allows us to simulate systems with billions of discontinuity points on our fractal curve, dual to the decaying turbulence. For the vorticity correlation function, we obtain quantum scaling laws with regime changes between effective indexes drifting with the logarithm of scale. The traditional description of turbulence with fractal or multifractal scaling laws does not apply here. In particular, the effective fractal dimension of our curve is a function of the scale. The measured conditional probabilities of fluctuating variables are smooth functions of the logarithm of scale, with statistical errors being negligible at our large number of random samples $T = 167,772,160$ with $N = 200,000,000$ points on a fractal curve. The quantum jumps arise from the analytical distribution of scaling variable $X = \frac{\cot^2(\pi p/q)}{q^2}$ where $\frac{p}{q}$ is a random fraction. This distribution is related to the totient summatory function and is a discontinuous function of $X$. In particular, the energy spectrum has quantum levels on top of a continuous background.

**Keywords:** turbulence; fractal; anomalous dissipation; fixed point; velocity circulation; numerical simulations; GPU; loop equations

---

## 0. Introduction

In 1993, one of us suggested [1] nonperturbative approach to Navier-Stokes turbulence based on the loop equations previously introduced in the gauge theory [2,3].

The Wilson loop average for the turbulence

$$\Psi[\gamma, C] = \left\langle \exp\left( \frac{\iota\gamma}{\nu} \oint d\theta \vec{C}'(\theta) \cdot \vec{v}(\vec{C}(\theta)) \right) \right\rangle_{init} \tag{1}$$

treated as a function of time and a functional of the periodic function $C : \vec{r} = \vec{C}(\theta);\ \theta \in (0, 2\pi)$ (not necessarily a single closed loop), satisfies certain functional equation which was defined and studied in these papers.

The averaging $\langle\rangle_{init}$ corresponds to averaging over an ensemble of solutions with different initial data. One does not need to specify this ensemble to obtain the Hopf or loop equation.

We are not going into details of the loop technology but shall investigate the solution found in the recent paper [4].

## 1. The theory

This solution has the following form

$$\Psi[\gamma, C] = \left\langle \exp\left( \frac{\iota\gamma}{\nu} \oint d\theta \vec{C}'(\theta) \cdot \vec{P}(\theta) \right) \right\rangle_F \tag{2}$$

The averaging $\langle\rangle_F$ corresponds to averaging over an ensemble of solutions for $\vec{F}(\theta)$ to be specified below.

This periodic vector function $\vec{P}(\theta)$ also depends of time as follows

$$\vec{P}(\theta, t) = \sqrt{\frac{\nu}{2(t + t_0)}} \frac{\vec{F}(\theta)}{\gamma}; \tag{3}$$

$$\tag{4}$$

The time-independent function $\vec{F}(\theta)$ satisfies two singular equations

$$(\Delta\vec{F})^2 = 1; \tag{5a}$$

$$(2\vec{F} \cdot \Delta\vec{F} - \iota\gamma)^2 + \gamma^2 = 4\vec{F}^2; \tag{5b}$$

$$\Delta\vec{F} = \vec{F}(\theta + 0) - \vec{F}(\theta - 0); \tag{5c}$$

$$\vec{F} = \frac{\vec{F}(\theta + 0) + \vec{F}(\theta - 0)}{2}; \tag{5d}$$

This equation describes a fixed point regime of the loop equations for decaying turbulence. No approximations have been made so far.

This $\vec{F}(\theta)$ is a fractal curve in complex 3D space, with discontinuity $\Delta\vec{F}$ at every angle $\theta$.

The solution in [4] builds such a curve as a limit of the polygon with $N \to \infty$ vertexes. The positions $\vec{F}_k = \vec{F}(2\pi k/N)$ of these vertices satisfy the recurrent equation

$$\left(\vec{F}_{k+1} - \vec{F}_k\right)^2 = 1; \tag{6a}$$

$$\left(\vec{F}_{k+1}^2 - \vec{F}_k^2 - \iota\gamma\right)^2 + \gamma^2 = \left(\vec{F}_{k+1} + \vec{F}_k\right)^2 \tag{6b}$$

There could be many solutions of such equations in $d > 2$ dimensions, as these are two complex equations relating $d$ complex parameters in $\vec{F}_{k+1}$ to those in $\vec{F}_k$.

Thus, there are $d - 2$ complex parameters undetermined in every step $\vec{q}_k = \vec{F}_{k+1} - \vec{F}_k$. This degeneration makes this a Markov chain or a random walk where each step depends only on a current position plus some arbitrary (random) parameters.

### 1.1. The Euler ensemble and its continuum limit

In particular, the following solution, found in [4], has some acceptable physical properties. We call it the Euler ensemble.

$$\vec{F}_k = \frac{1}{2} \csc\left(\frac{\beta}{2}\right) \hat{\Omega} \cdot \left\{\cos(\alpha_k), \sin(\alpha_k)\vec{w}, i\cos\left(\frac{\beta}{2}\right)\right\}; \tag{7}$$

$$\hat{\Omega} \in O(d); \tag{8}$$

$$\beta = \frac{2\pi p}{q}; \tag{9}$$

$$0 < p < q < N; \tag{10}$$

$$\alpha_k = \beta \sum_0^{k-1} \sigma_l; \tag{11}$$

$$\sigma_k^2 = 1; \tag{12}$$

$$\sum_0^{N-1} \sigma_l = qr; \tag{13}$$

$$-N \leq qr \leq N; \tag{14}$$

$$(N - rq) \pmod 2 = 0; \tag{15}$$

Here $p, q, r, N$ are positive integers, $\hat{\Omega}$ is a random rotation matrix in $O(d)$, and $\vec{w} \in \mathbb{S}^{d-3}$ is a unit vector in $d - 2$ dimensional space $\mathbb{R}^{d-2}$.

This paper only considers the three-dimensional space $d = 3$ where $\vec{w} = \pm 1$. This sign factor can be absorbed into a redefinition of $\sigma_l$.

This sequence with arbitrary signs $\sigma_k = \pm 1$ subject to periodicity constraint (13) solves recurrent equation (6) independently of $\gamma$.

This ensemble consists of random $p, q, r, N, \sigma_0 \ldots \sigma_{N-1}$ subject to periodicity conditions; we called it an Euler ensemble.

The **ergodic hypothesis** of [4] states that each element of the Euler ensemble enters with equal weight. This includes every set of $\sigma_l$ satisfying periodic boundary condition (13), every pair of coprime numbers $0 < p < q < N, \gcd(p, q) = 1$ and every $q, r, N$ with $(N - qr) \pmod 2 = 0$.

The number theory counts these states using Euler totients.

A unique property of this solution is that the velocity circulation is real in this solution, as it should, despite the complex values of $\vec{F}_k$.

This cancellation of the imaginary part happens because of the closure (i.e., periodicity) of both loops $\vec{C}(\theta), \vec{F}(\theta)$

$$\oint d\theta \vec{C}'(\theta) \cdot \vec{F}(\theta) = - \oint d\theta \vec{C}(\theta) \cdot \vec{F}'(\theta);; \tag{16}$$

$$\sum_k \Delta\vec{C}_k \cdot \vec{F}_k = - \sum_k \vec{C}_k \cdot \Delta\vec{F}_k; \tag{17}$$

The discontinuity (i.e., the polygon side) is a real unit vector :

$$\vec{q}_k = \Delta\vec{F}_k = \sigma_k \hat{\Omega} \cdot \{-\sin\delta_k, \vec{w}\cos\delta_k, 0\}; \tag{18}$$

$$\delta_k = \alpha_k + \frac{\beta\sigma_k}{2}; \tag{19}$$

The sum over all different configurations of the Euler ensemble with exponential weight $\exp(-\mu N)$ is called a grand canonical ensemble, using the language of statistical physics. The number theory counting using Euler totients [5] produces the following general formula [4]

$$Z(\mu) = \sum_N Z(N, N) e^{-\mu N}; \tag{20}$$

$$\langle F(p, q, r, N) \rangle_{\mathcal{E}(\mu)} = \frac{\sum_N e^{-\mu N} \sum_{\substack{2 < q < N}} \sum_{\substack{p=1 \\ (p,q)=1}}^{q-1} \sum_{\substack{r \\ 2|(N-qr)}} F(p, q, r, N)}{Z(\mu)}; \tag{21}$$

At the critical point $\mu \to 0$, the even $N$ dominate, with the following result

$$Z(\mu) \to \frac{1}{2} \frac{9}{2\sqrt{2}\pi^2 \mu^{5/2}} \tag{22}$$

The enstrophy, defined as an expectation value of the square of local vorticity, reduces in this theory to the following time-dependent formula [4]

$$\left\langle \vec{\omega}(\vec{0})^2 \right\rangle = \frac{1}{(t + t_0)^2} \left\langle 2^{-N-4} S(q) \left(N - q^2 r^2\right) \binom{N}{(N + qr)/2} \right\rangle_{\mathcal{E}(\mu)}; \tag{23}$$

$$S(q) = \sum_{\substack{p=1 \\ (p,q)=1}}^{q-1} \cot^2\left(\frac{\pi p}{q}\right); \tag{24}$$

We have found the method to compute this sum of squares of cotangent of all fractions of $\pi$ with a given denominator analytically. This result was quoted in [4] and then generalized to arbitrary powers of cotangent.

This method is described in Appendix A

*1.2. Small Euler ensemble at large N*

At a fixed large value of $q$, the CDF of $x = \beta/(2\pi)$ is given by the ratio of the Euler totient functions, asymptotically equal to $x$ at large $q$

$$\mathbb{P}(\beta < 2\pi x \mid q) = \frac{\varphi(xq)}{\varphi(q)} \to x; \tag{25}$$

The CDF of $q$ is proportional to the totient summatory function

$$\Phi(n) = \sum_{k=1}^{n} \varphi(k) \to \frac{6}{\pi^2} n^2 \tag{26}$$

Thus, the normalized CDF of $y = q/N$

$$\mathbb{P}(q < yN) \to y^2; \tag{27}$$

$$\frac{dW(y)}{dy} \to 2y\theta(y)\theta(1-y); \tag{28}$$

As we shall see, rather than $\beta$, we would need an asymptotic distribution of a scaling variable

$$X(p,q) = \frac{1}{q^2} \cot^2\left(\frac{\pi p}{q}\right) \to \frac{1}{\pi^2 p^2} \tag{29}$$

The $\beta N$ variable can be related to $y$ and $X(p,q)$

$$\beta N = 2N \operatorname{arcCot}(q\sqrt{X}) \to \frac{2}{y\sqrt{X}}; \tag{30}$$

This distribution for $X(p,q)$ at fixed $q \sim N \to \infty$ can be found analytically, using newly established relations for the cotangent sums (see Appendix in [4]). Asymptotically, at large $q$ these relations read

$$\langle X^n \rangle \equiv \lim_{q\to\infty} \frac{1}{q} \sum_{\substack{p=1 \\ (p,q)}}^{q-1} X(p,q)^n = \delta_{n,0} + \frac{2\pi^{-2n}\zeta(2n)}{(2n+1)\zeta(2n+1)} \tag{31}$$

This relation can be transformed further as

$$\langle X^n \rangle = \begin{cases} 1 & \text{if } n = 0 \\ \dfrac{2\displaystyle\sum_{k=1}^{\infty} \varphi(k)k^{-(2n+1)}}{(2n+1)\pi^{2n}} & \text{if } n > 0 \end{cases} \tag{32}$$

The Mellin transform of these moments leads to the following singular distribution

$$\langle X^n \rangle = \int_0^\infty W'(X)\, dX\, X^n;$$  (33)

$$W'(X) = (1 - \alpha)\delta(X - 0^+) + \pi\Phi\left(\left\lfloor \frac{1}{\pi\sqrt{X}} \right\rfloor\right);$$  (34)

$$\alpha = \int_0^\infty \pi\Phi\left(\left\lfloor \frac{1}{\pi\sqrt{X}} \right\rfloor\right)\, dX = \pi\sum_{k=1}^\infty \varphi(k)\frac{2k+1}{k^2(k+1)^2} = 0.387153$$  (35)
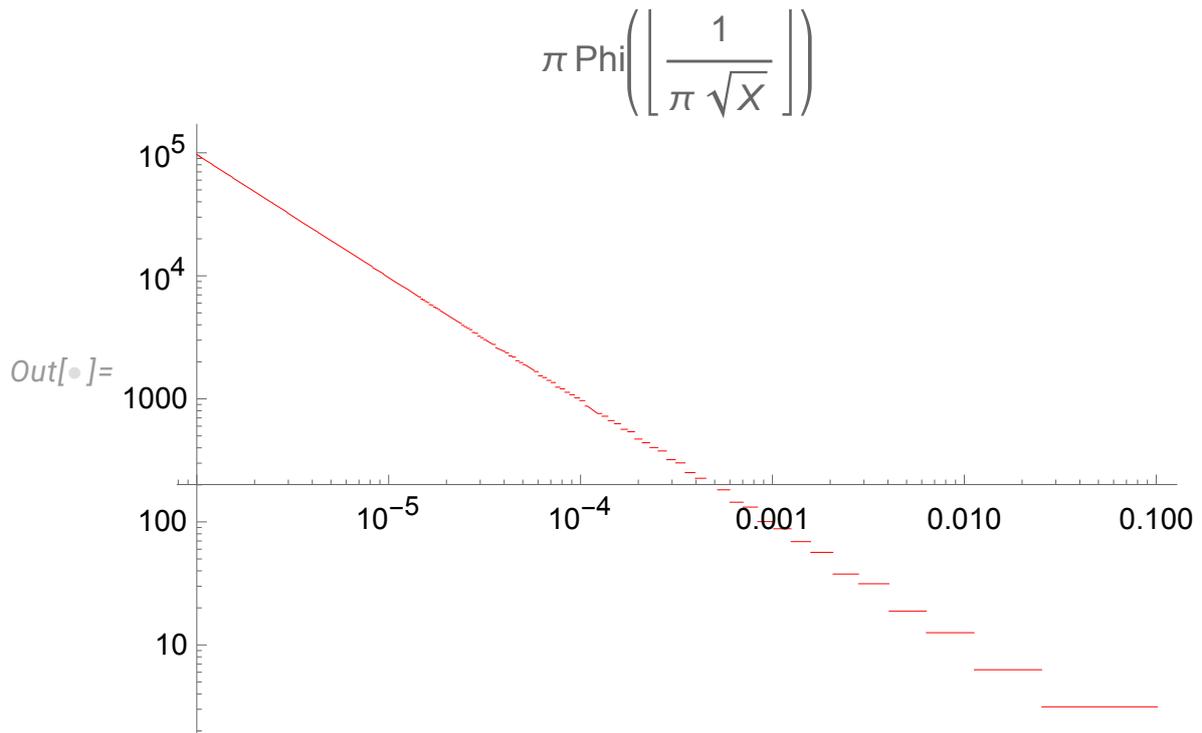
where $\Phi(n)$ is the totient summatory function



**Figure 1.** The log-log-plot of the totient summatory function. It is takes a constant value $\frac{\varphi(k)}{\pi}$ at each internal $\frac{1}{\pi^2(k+1)^2} < X < \frac{1}{\pi^2 k^2}$ with positive integer $k$. The convergent integral $\alpha = 0.387153$ of this piecewise constant function reduces to the sum of these constant values times the widths of the intervals.

The first term $(1 - \alpha)\delta(X - 0^+)$ accumulates the contribution from the vicinity of the finite lower limit of $X$

$$X_{min} = X\left(\lfloor q/2 \rfloor - 1, q\right) \to \frac{\pi^2}{q^4} = 0^+$$  (36)

This lower limit tends to zero, which leaves some part of the delta function out of the support of probability distribution. Normalizing the base moment $\langle X^0 \rangle = 1$ yields the correction factor $(1 - \alpha)$ in front of the delta function. Therefore, the $\alpha$ part of the delta function in the normalized distribution falls below the lower limit, with the remaining part $(1 - \alpha)$ staying in the support of the distribution.

The upper limit of $X$

$$X_{max} = X\left(q - 1, q\right) \to \frac{1}{\pi^2}$$  (37)

Our distribution (33) is consistent with this upper limit, as the argument $\left\lfloor \frac{1}{\pi\sqrt{X}} \right\rfloor$ becomes zero at $X\pi^2 > 1$. It is plotted in Figure 1, except for the delta function part at infinitesimal $X$.

Once we are zooming into the tails of the $p, q$ distribution, we also must recall that

$$W(y) = \Phi\left(\lfloor Ny \rfloor\right); \tag{38}$$

$$W'(y) = \sum_{q=2}^{\infty} \delta\left(y - \frac{q}{N}\right) \varphi(q) \tag{39}$$

The variable $s$ also has a nontrivial distribution in the small Euler ensemble with even $N \to \infty$, with a dominant contribution at $s = 0$ as conjectured in [4] and proven in [6]

$$\frac{dW(s)}{ds} = \delta(s) + O\left(N^{-1/2}\right) \tag{40}$$

## 2. The big Euler ensemble as a Markov process

### 2.1. Numerical Simulations

We took $N = 200,000,000$ and generated $T = 167,772,160$ random data samples for $\beta, \vec{\omega}_n \cdot \vec{\omega}_m, \left|\vec{S}_{n,m} - \vec{S}_{m,n}\right|$ with randomly chosen $0 \leq n < m < N$.

We devised a fast Python/CPP algorithm for this simulation, described in Appendix B. We ran it on an NYUAD cluster Jubail, which took about a week altogether, for three GPU runs and three CPU runs, which we all pooled into one dataset and binned by $2^{20}$ intervals by the ranking of $\log\cot^2(\beta/2)$. Two datasets were created, one for each sign of $\vec{\omega}_n \cdot \vec{\omega}_m$.

We used a recursive rank-binning algorithm based on the standard library function *nth-element*, which finds the median of unordered data by partial sorting in place in $O(N)$ time. We recursively applied this function in parallel to each of the two halves of the array of data. The CPU time of this recursive parallel code is linear, as the geometric series $N + N/2 + N/4 + \ldots$ adds up to $2N$.

Each interval contained only $n = 80 \pm 1$ data points $\{n, \bar{c}, \Delta c, \bar{s}, \Delta s, \bar{w}, \Delta w\}$, where

$$c = 2\log\left|\cot(\beta/2)\right|; \tag{41}$$

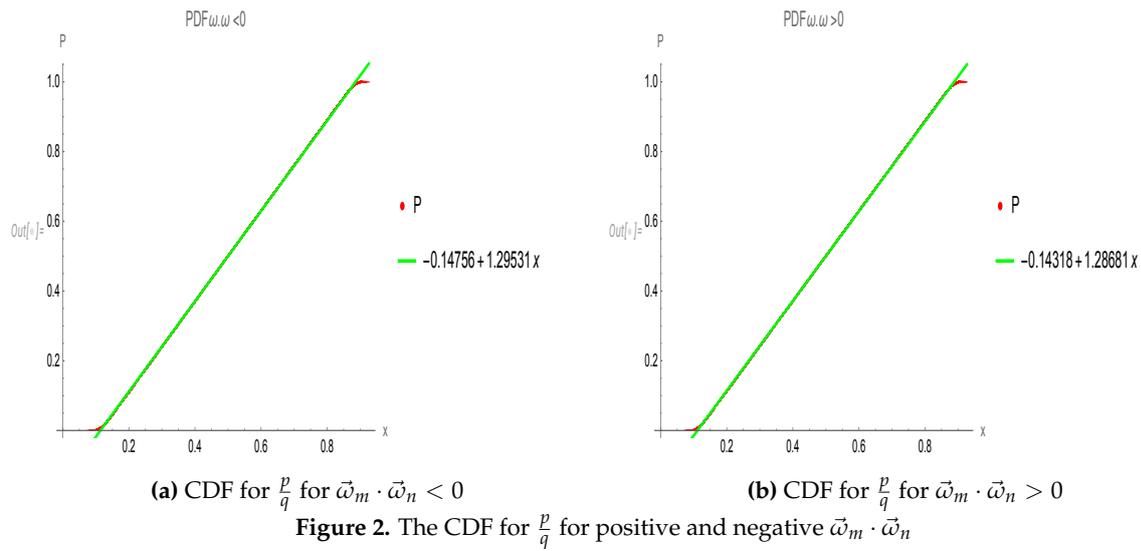$$s = \log\left|\vec{S}_{n,m} - \vec{S}_{m,n}\right|; \tag{42}$$

$$w = \log\left|\vec{\omega}_n \cdot \vec{\omega}_m\right|; \tag{43}$$

and $\bar{x}, \Delta x$ denotes mean and standard deviation. Later, we used these data with Gaussian errors for the statistical analysis. We used logarithms because the values varied by several orders of magnitude, preventing statistical analysis.

The code was optimized to take $O(N^0)$ RAM, so the CPU and GPU resources of all 200 nodes were used to collect statistics. This optimization allows us to simulate astronomical ensembles of random fractions without hitting memory limits.

The first thing we measured was the distribution of $x = \frac{\beta}{2\pi} = \frac{p}{q}$, which we know in advance to be uniform except for small regions of $x \sim 1/N, 1 - x \sim 1/N$, where quantization plays the role.

The measured distribution supports this hypothesis (see Figure 2).

**(a)** CDF for $\frac{p}{q}$ for $\vec{\omega}_m \cdot \vec{\omega}_n < 0$



**(b)** CDF for $\frac{p}{q}$ for $\vec{\omega}_m \cdot \vec{\omega}_n > 0$

**Figure 2.** The CDF for $\frac{p}{q}$ for positive and negative $\vec{\omega}_m \cdot \vec{\omega}_n$

The CDF is linear, supporting the uniform distribution of $\frac{p}{q}$ except for the endpoints $(0, 1)$.

However, the critical phenomena in our system are related to these two endpoints where the scaling factor $\cot^2(\beta/2)$ grows to $\sim q^2$. The uniform distribution of $\beta$ corresponds to the variable $X = q^{-2} \cot^2(\beta/2)$ distributed at large $q$ as

$$\frac{dW_0(X)}{dX} \propto q^{-1} X^{-3/2} \tag{44}$$

The exact asymptotic distribution for the same variable has the form (34), found in the previous section.

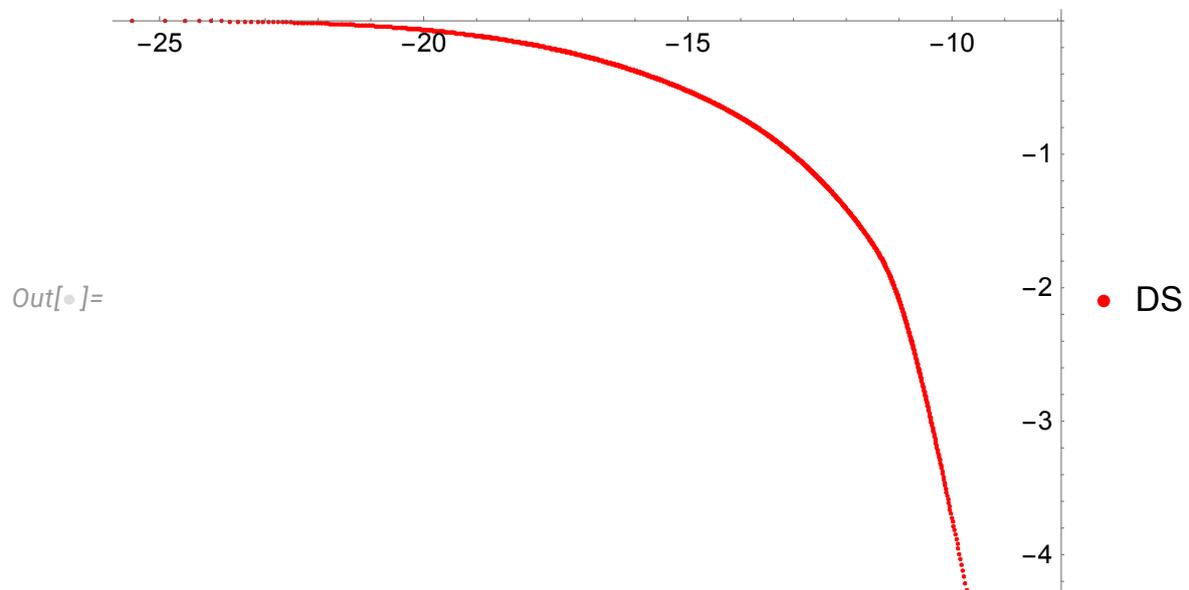## 2.2. Scaling variables in continuum limit



**Figure 3.** Log-log plot of CDF tail for $\left| \vec{S}_{n,m} - \vec{S}_{m,n} \right| / q$. The curve is nonlinear in log scale, contradicting the simple fractal scaling laws.
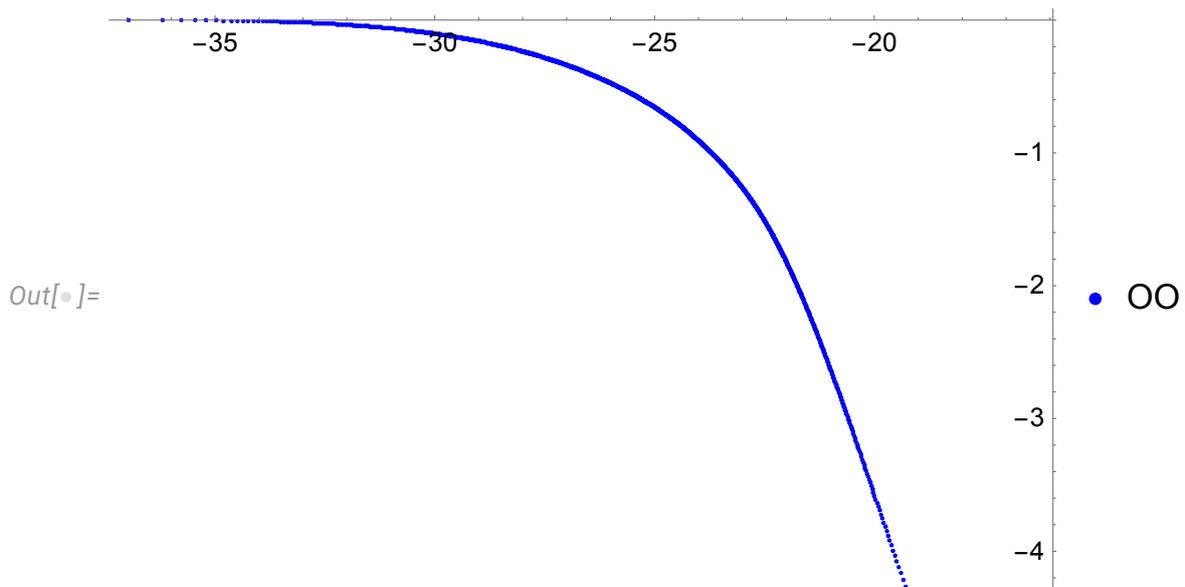
*Out[•]=*

**Figure 4.** Log-log plot of CDF tail for $|\vec{\omega}_m \cdot \vec{\omega}_n| / q^2$. The curve is nonlinear in log scale, contradicting the simple fractal scaling laws.

This quantum distribution is too complicated to reproduce in numerical simulation, but we measured distributions of other variables that behave smoother.

We studied the distribution of two variables involved in the correlation function (see next section)

$$DS = q^{-1} \left| \vec{S}_{n,m} - \vec{S}_{m,n} \right|; \tag{45}$$

$$WW = q^{-2} \vec{\omega}_m \cdot \vec{\omega}_n; \tag{46}$$

We multiplied both variables by powers of $q$ to make them finite in the statistical limit $q \sim N \to \infty$.

The distributions of these variables are shown in Figure 4, Figure 3.

The variable $DS$ measures the mean Euclidean distance between two random points on a curve in the embedding 3D complex space.

Let us study the moments of this distribution. Integrating by part in the standard definition, we have

$$\langle DS^n \rangle = \int W'(DS) DN^n d(DS) = n \int (1 - W(DS)) DS^{n-1} dDS =$$

$$\int \exp\left(n\xi + T(\xi) + \log n\right) d\xi; \tag{47}$$

$$T(\xi) = \log\left(\left(1 - W\left(\exp\left(\xi\right)\right)\right)\right); \tag{48}$$

Our data was fit perfectly by 20 Chebyshev polynomials for this $T(\xi)$ (see [7] and Figure 5).
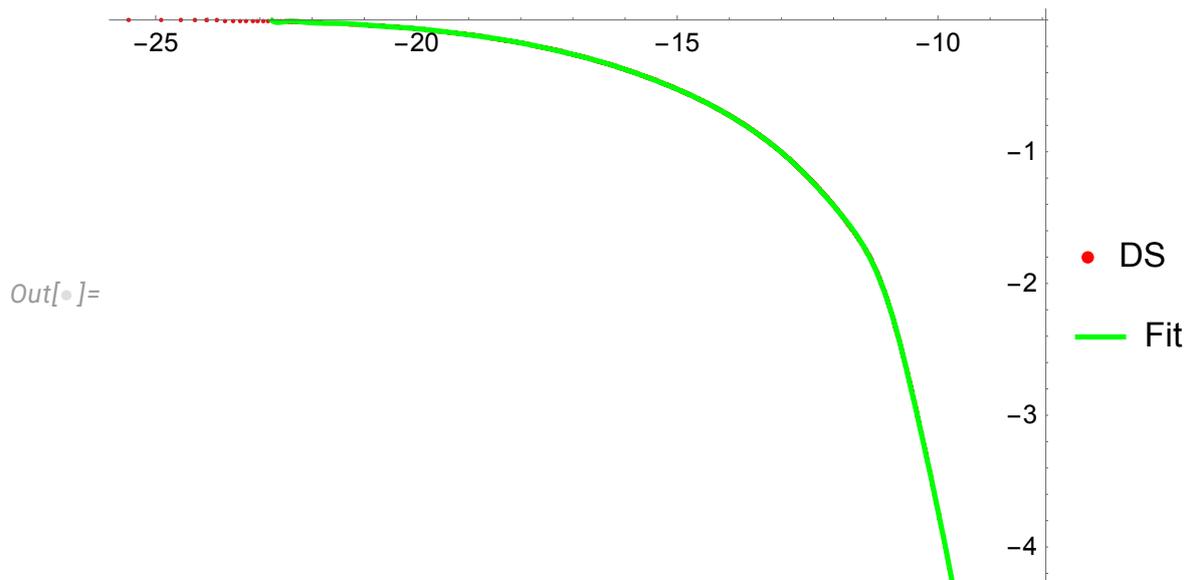
$Out[\bullet]=$

**Figure 5.** Fit of $\log(1 - CDF(DS))$ with 20 Chebyshev polynomials of $\log DS$ rescaled to $-1, 1$ interval. The small statistics data at large negative $\log DS$ was discarded

The definition of effective index can be given as

$$\eta(n) = \frac{\log \langle (DS)^n \rangle}{\log \langle DS \rangle};$$ (49)

The saddle point computation of this integral gives the following parametric equations

$$n = -T'(\xi);$$ (50)

$$\log \langle DS^n \rangle \to L_n(\xi) = (n\xi + T(\xi) + \log n)_{n=-T'(\xi)};$$ (51)

$$L_1 = (\xi_1 + T(\xi_1))_{1=-T'(\xi_1)};$$ (52)

$$\eta(-T'(\xi)) = \frac{L_n(\xi)}{L_1}$$ (53)

This function was computed in [7], and here is the plot in Figure 6. It is numerically close to a linear function , though it is analytically nonlinear.
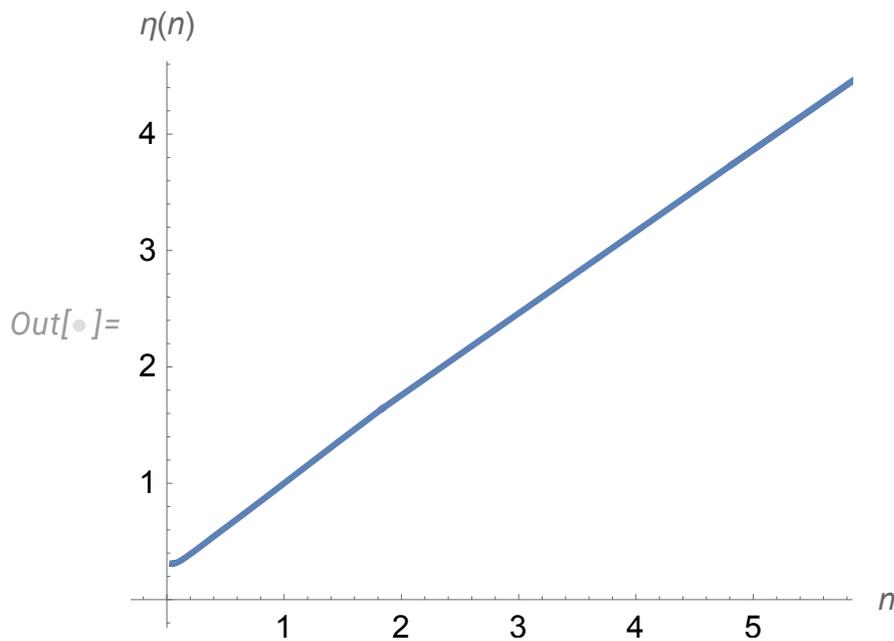
**Figure 6.** The multifractal dimension $\eta(n)$ of moments $\langle DS^n \rangle$. The curve is numerically close to a linear function.

We also study some quantum fractal phenomena in the next section. The observable variables are distributed with quantum jumps, not fitting the multifractal framework.

There is another variable that depends on two spins at two different points $\sigma_n, \sigma_m, n < m$

$$Y(p,q,\sigma) = -\sigma_n \sigma_m X(p,q) \tag{54}$$

The even moments $\left\langle Y^{2l} \right\rangle$ of the distribution of $Y$ are the same as those for $X$, but the odd moments are different:

$$\left\langle Y^{2l+1} \right\rangle = - \left\langle X^{2l+1} \langle \sigma_n \sigma_m \rangle_\sigma \right\rangle_\varepsilon \tag{55}$$

The analytic computation of the constrained average over sigmas by *Mathematica*® yields

$$\langle \sigma_n \sigma_m \rangle_\sigma = -\frac{N - q^2 r^2}{(N-1)N} \tag{56}$$

We find the following result for the distribution of $Y$ on the whole real axis

$$f_Y(Y \mid N, q, r) = \left( \frac{1}{2} + \text{sign}\, Y \frac{N - q^2 r^2}{2(N-1)N} \right) f_X(|Y|) \tag{57}$$

Asymptotically, at $s = qr/N \sim 1$ we have

$$\frac{dW(Y \mid s)}{dY} \to \frac{1 - \text{sign}(Y)s^2}{2} \left( (1-\alpha)\delta(|Y| - 0^+) + \pi\Phi \left( \left\lfloor \frac{1}{\pi\sqrt{|Y|}} \right\rfloor \right) \right) ; \tag{58}$$

Due to kinematical restrictions $|y| < 1$, our probably stays positive. At $r = 0$, we have a special case

$$\frac{dW(Y \mid 0)}{dY} \to \frac{N + \text{sign}(Y)}{2N} \left( (1-\alpha)\delta(|Y| - 0^+) + \pi\Phi \left( \left\lfloor \frac{1}{\pi\sqrt{|Y|}} \right\rfloor \right) \right) ; \tag{59}$$

### 3. Vorticity correlation

The simplest observable quantities we can extract from the loop functional are the vorticity correlation functions [8], corresponding to the loop $C$ backtracking between two points in space $\vec{r}_1 = 0, \vec{r}_2 = \vec{r}$, (see [4] for details and the justification). The vorticity operators are inserted at these two points.

The correlation function reduces to the following average over the big Euler ensemble $\mathbb{E}$ of our random curves in complex space [4]

$$\left\langle \vec{\omega}(\vec{0}) \cdot \vec{\omega}(\vec{r}) \right\rangle =$$

$$\frac{1}{4(t+t_0)^2} \int_{O(3)} d\Omega \sum_{0 \leq n < m < N} \left\langle \vec{\omega}_m \cdot \vec{\omega}_n \exp \left( \imath \vec{\rho} \cdot \Omega \cdot \left( \vec{S}_{n,m} - \vec{S}_{m,n} \right) \right) \right\rangle_{\mathbb{E}}; \tag{60}$$

$$\vec{S}_{n,m} = \frac{\sum_{k=n}^{m-1} \vec{F}_k}{m - n \pmod{N}}; \tag{61}$$

$$\vec{\omega}_k = \left\{ 0, 0, \frac{\imath \sigma_k}{2} \cot \left( \frac{\beta}{2} \right) \right\}; \tag{62}$$

$$\vec{\rho} = \frac{\vec{r}}{2\sqrt{\nu(t+t_0)}}; \tag{63}$$

$$\langle X[\sigma...] \rangle_{p,q,r} \equiv \frac{\sum_{\mathbb{E}} X[\sigma...] \delta[qr - \sum \sigma]}{\sum_{\mathbb{E}} \delta[qr - \sum \sigma]}; \tag{64}$$

Integrating the global rotation matrix $O(3)$ is part of the ensemble averaging.

These formulas simplify in the Fourier space, corresponding to the correlation of $\vec{\omega}(\vec{k}) \cdot \vec{\omega}(-\vec{k})$.

$$\left\langle \vec{\omega}(\vec{k}) \cdot \vec{\omega}(-\vec{k}) \right\rangle \propto \nu^{\frac{3}{2}} (t+t_0)^{-\frac{1}{2}} \sum_{0 \leq n < m < N}$$

$$\left\langle \vec{\omega}_m \cdot \vec{\omega}_n \int_{O(3)} d\Omega \delta^3 \left( \Omega \cdot \left( \vec{S}_{n,m} - \vec{S}_{m,n} \right) - \vec{k} \sqrt{\nu(t+t_0)} \right) \right\rangle_{\mathbb{E}} \tag{65}$$

This three-dimensional delta function disappears if we introduce the above PDF for $Y = q^{-2} \vec{\omega}_m \cdot \vec{\omega}_n$ and conditional probability for the second variable $Z = \frac{|\vec{S}_{n,m} - \vec{S}_{m,n}|}{q}$ given value of $Y$ below some threshold $y$. We rely on $r = 0$ and even $N$ to dominate the partition function, as it was conjectured in [4] and proven in [6]. We also assume the probability $\mathbb{P}(Z \mid y)$ for the second scaling variable $Z$, conditional by $|Y| < y$, to be independent of the sign of $Y$. The heuristic argument is that the sign of $Y$ depends only on $\sigma_n \sigma_m$, only two out of $N$ variables $\sigma_l$, which becomes statistically insignificant at large $N$.

Our large-scale numerical simulation confirms this heuristic argument for $N = 2 * 10^8$. In this case

$$\left\langle \vec{\omega}(\vec{k}) \cdot \vec{\omega}(-\vec{k}) \right\rangle \propto \nu^{\frac{3}{2}} (t + t_0)^{-\frac{1}{2}} \int dW(q) q^4$$

$$\int \mathbb{P}(Y; 0) Y \, dY \int d\Omega \int dZ \mathbb{P}(Z \mid Y) \delta^3 \left( \Omega \cdot \{0, 0, Zq\} - \vec{k} \sqrt{\nu(t + t_0)} \right) =$$

$$\frac{\sqrt{\nu}}{\vec{k}^2 (t + t_0)^{\frac{3}{2}}} \int dW(q) q \int dy \, \mathbb{P}(y \mid 0) y \frac{d\mathbb{P}\left( \frac{|\vec{k}| \sqrt{\nu(t+t_0)}}{q} \mid y \right)}{dy} =$$

$$\frac{-\sqrt{\nu}}{\vec{k}^2 (t + t_0)^{\frac{3}{2}}} \int dW(q) q \int dy \, \mathbb{P} \left( \frac{|\vec{k}| \sqrt{\nu(t+t_0)}}{q} \mid y \right) \partial_y \left( \mathbb{P}(y \mid 0) y \right) ; \tag{66}$$

$$\frac{dW(q)}{dq} = \Phi(q) \tag{67}$$

The distribution $P(y \mid 0)$ was found above, in (59) by the number theory methods. We are using the derivative

$$\partial_y (\mathbb{P}(y \mid 0) y) = \pi \frac{\partial \left( |y| \Phi \left( \left\lfloor \frac{1}{\pi \sqrt{|y|}} \right\rfloor \right) \right)}{\partial y} =$$

$$\pi \Phi \left( \left\lfloor \frac{1}{\pi \sqrt{|y|}} \right\rfloor \right) + \frac{2}{\pi^3} \sum_{n=1}^{\infty} \frac{\varphi(n)}{n^5} \delta \left( y - \frac{1}{\pi^2 n^2} \right) \tag{68}$$

The resulting formula for the correlation function

$$\left\langle \vec{\omega}(\vec{k}) \cdot \vec{\omega}(-\vec{k}) \right\rangle \propto \frac{\sqrt{\nu}}{\vec{k}^2 (t + t_0)^{\frac{3}{2}}} \int dq \, q \Phi(q)$$

$$\left( \frac{2}{\pi^3} \sum_{n=1}^{\infty} \frac{\varphi(n)}{n^5} \mathbb{P} \left( Z \mid Y < \frac{1}{\pi^2 n^2} \right) + \pi \sum_{n=1}^{\infty} \Phi(n+1) \int_{\frac{1}{\pi^2 (n+1)^2}}^{\frac{1}{\pi^2 n^2}} dy \mathbb{P}(Z \mid Y < y) \right) ; \tag{69}$$

$$Z = \frac{|\vec{k}| \sqrt{\nu(t + t_0)}}{q} ; \tag{70}$$

We measured conditional probability in our data set of $167,772,160$ samples for $N = 200,000,000$ (see Figure 7).
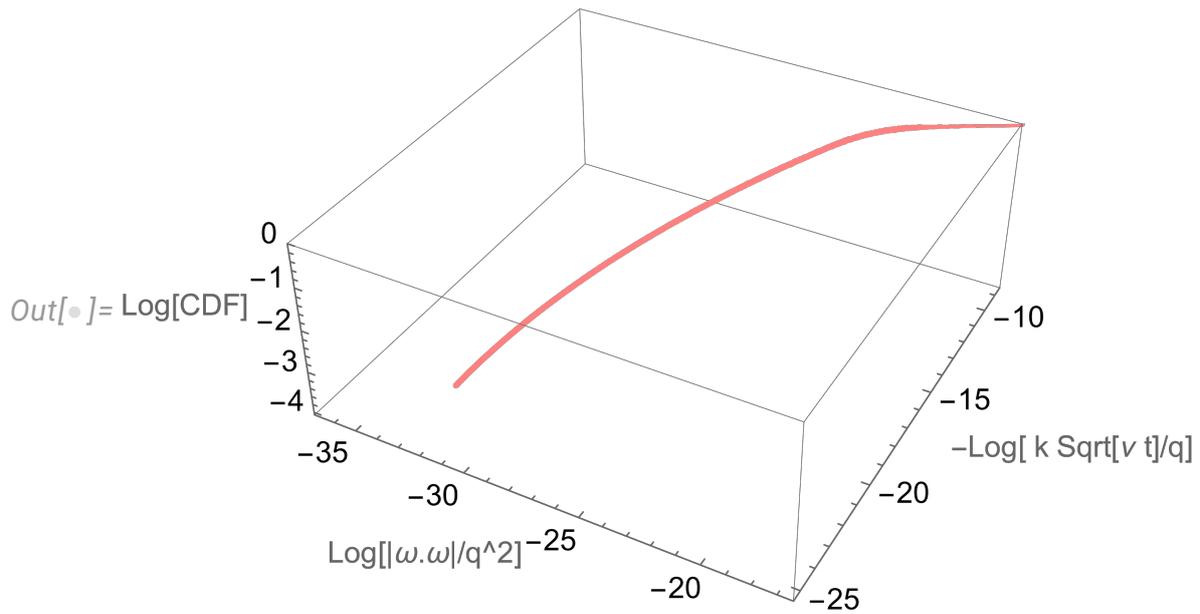
**Figure 7.** 3D Plot of the conditional probability $\mathbb{P}\left(\log DS \mid \log WW < y\right)$ in a triple log scale. The data (red dots) was averaged over 1000 samples sorted by $\log WW$.

The conditional probability $\mathbb{P}\left(\log DS \mid \log WW < y\right)$ was defined as a rate of events with given $\log DS$ and $\log WW < y$. It is shown as a 3D plot on the log-log-log scale in Figure 7.

In a classical fractal system, one would expect to observe the two-dimensional surface $\mathbb{P}\left(\log Z \mid \log WW < Y\right)$ in a three-dimensional space $Y, Z, \mathbb{P}$.

Instead, we found a thin, smooth line in three dimensions, which can be fitted as a parametric equation

$$\log Y = a + b \log Z; \tag{71}$$

$$\log Z = f(x); \tag{72}$$

$$\log \mathbb{P}\left(\log Z \mid \log WW < Y\right) = -x; \tag{73}$$

$$f(x) \approx f_0 + f_1 x + f_2 x^2 + f_3 x^3; \tag{74}$$

Projections of this line on three planes reveal its simplicity.
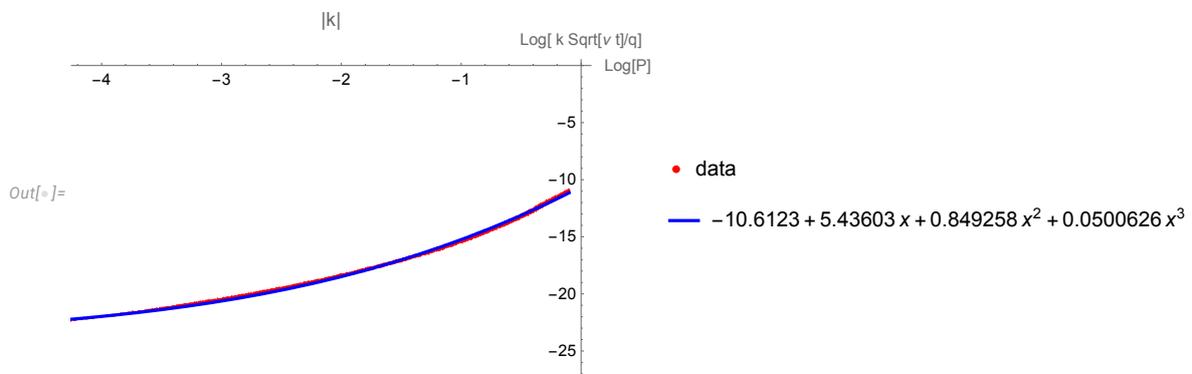The relation between $\log Z, \log \mathbb{P}$ is shown in Figure 8.



**Figure 8.** The relation of $\log Z, \log \mathbb{P}$ (red dots) is fitted by a third-degree polynomial (blue)

The fit parameter table for $f(x)$ is

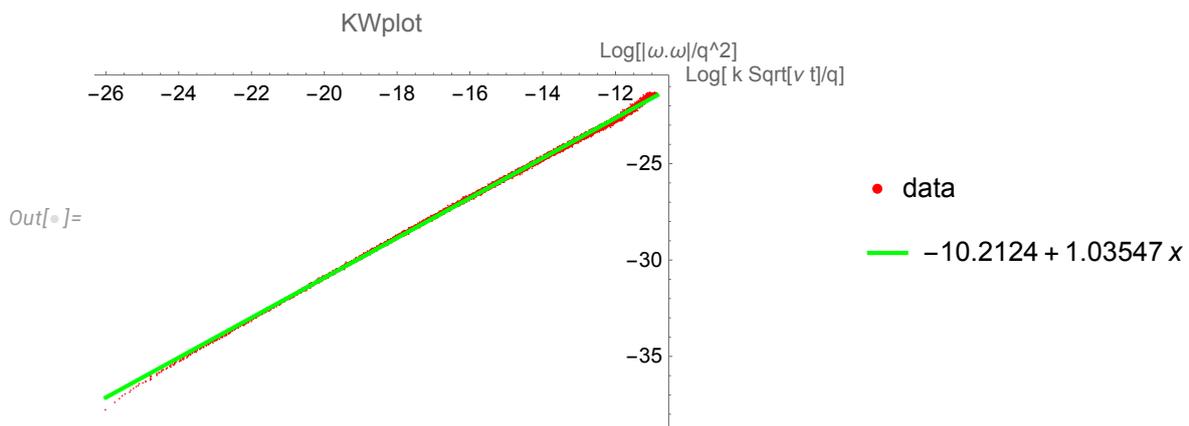|       | Estimate    | Standard Error | t-Statistic | P-Value |
|-------|-------------|----------------|-------------|---------|
| $f_0$ | $-10.6123$  | 0.0030473      | $-3482.54$  | 0.      |
| $f_1$ | 5.43603     | 0.00542859     | 1001.37     | 0.      |
| $f_2$ | 0.849258    | 0.00224021     | 379.097     | 0.      |
| $f_3$ | 0.0500626   | 0.000234038    | 213.908     | 0.      |

$$(75)$$



**Figure 9.** Direct scaling relation between $WW$ and $DS$, ads fit to the log-log of joint distribution

If we directly plot $Y(Z)$; we get a straight line, corresponding to a scaling law (see Figure 9). The fitting table for this linear relation $\log Y = a + b \log Z$ in the log-log scale is

|     | Estimate    | Standard Error | t-Statistic | P-Value |
|-----|-------------|----------------|-------------|---------|
| $a$ | $-10.2124$  | 0.0039544      | $-2582.54$  | 0.      |
| $b$ | 1.03547     | 0.00025789     | 4015.16     | 0.      |

$$(76)$$

So, a scaling law is hidden inside the discontinuous distributions induced by Euler totients.

$$Y = AZ^b; \tag{77}$$

$$A = \exp(a) = 0.0000367128; \tag{78}$$

The $k$- dependence of the vorticity correlation function is equivalent to a **quantized** energy spectrum on top of a continuous one

$$\frac{dE(k,t)}{dk} \propto k^2 \langle \vec{v} \cdot \vec{v}(k) \rangle \propto \langle \vec{\omega} \cdot \vec{\omega}(k) \rangle =$$

$$\frac{\nu^2}{\sqrt{\nu(t+t_0)}} G_N(K) H(K); \tag{79}$$

$$K = |\vec{k}| \sqrt{\nu(t+t_0)}; \tag{80}$$

$$G_N(K) = K^2 \int_{-\infty}^{0} e^{L-2f(L)} f'(L) \Phi\left( \left\lfloor e^{-f(L)} K \right\rfloor \right) dL =$$

$$K^2 \sum_{q=\lfloor K \exp(-f_0) \rfloor}^{N} \Phi(q) \int_{\log(K/(q+1))}^{\log(K/q)} dx \exp\left( -2x + f^{-1}(x) \right); \tag{81}$$

$$H(K) = \frac{2}{\pi^3} \sum_{n=1}^{\infty} \frac{\phi(n)}{n^5} \delta\left( AK^b - \frac{1}{n^2\pi^2} \right) + \pi \Phi\left( \left\lfloor \frac{1}{\sqrt{AK^b}\pi} \right\rfloor \right); \tag{82}$$

Here $w = f^{-1}(x)$ is a real positive solution of a cubic equation $f(w) = x$.

The function $G_N(K)$ in continuum limit $N \to \infty$ tends to a constant, independent of $K$, but growing with $N$

$$G_N(K) \propto N^4 \tag{83}$$

This work neglects such normalization factors, not affecting the energy spectrum.

The summation $\sum_n$ in H(K) at any given $K$ selects only one term with a quantized spectrum

$$K_n = \frac{1}{(An^2\pi^2)^{\frac{1}{b}}}, n = 1, 2, \dots; \tag{84}$$

$$|\vec{k}|_n = \frac{K_n}{\sqrt{\nu(t+t_0)}}; \tag{85}$$

The whole spectrum support lies at $K < K_1$; otherwise, both terms in $H(K)$ vanish. The critical region is $K \to 0$, which means the smaller and smaller wavevectors at larger and larger times. In the limit $K \to 0$

$$H(K) \propto \frac{1}{K^b}; \tag{86}$$

$$\frac{dE(k,t)}{dk} \propto \frac{N^4\nu^2}{|\vec{k}|^b(\nu(t+t_0))^{(b+1)/2}}; \tag{87}$$

This is not the whole story. There is a **maximal time** for any wavevector

$$t_{max}(k) = -t_0 + \frac{K_1^2}{\nu\vec{k}^2} \tag{88}$$

The decay stops at this wavelength after this moment, as $H(K > K_1) = 0$.

The quantum effects lead to peaks and jumps at finite $K = k\sqrt{\nu(t+t_0)}$. Only the average over a large range of the spectrum approaches the scaling law.

The integral over $\vec{k}$ leads to a simple power law for the energy dissipation rate

$$\mathcal{E} \propto 4\pi \int_0^\infty dk k^2 \langle \vec{\omega} \cdot \vec{\omega}(k) \rangle \propto \frac{\nu^2}{\sqrt{\nu(t+t_0)}} \int_0^\infty dk k^2 H\left(k\sqrt{\nu(t+t_0)}\right) \propto$$

$$\frac{\nu^2}{\sqrt{\nu(t+t_0)}^4} \int_0^{K_1} K^2 dK H(K) \propto (t+t_0)^{-2} \tag{89}$$

This law agrees with the exact computation of the same quantity made in [4]. Comparing coefficients in front of $(t+t_0)^{-2}$, one can restore the missing normalization factor in our energy spectrum.

$$\frac{dE(k,t)}{dk} = \langle \vec{\omega} \cdot \vec{\omega}(k) \rangle = \frac{35\pi^2}{3456\zeta(3)\mu^2} \frac{\nu^2}{\sqrt{\nu(t+t_0)}} \frac{H\left(|\vec{k}|\sqrt{\nu(t+t_0)}\right)}{4\pi \int_0^{K_1} K^2 dK H(K)} \tag{90}$$

where $\mu \sim \sqrt{\nu} \to 0$ is the chemical potential introduced in [4].

Finally, let us plot the universal function $H(K)$, using a width $\sigma = 10^{-3}$ for normal distribution instead of a pure delta function with $\sigma = 0$ (see Figure 10). The narrow peaks at the $K = K_n$ are not visible because of the very small heights of these peaks. The curve is made of steps with diminishing width as $K \to 0$, imitating on average the power law $K^{-b}$.
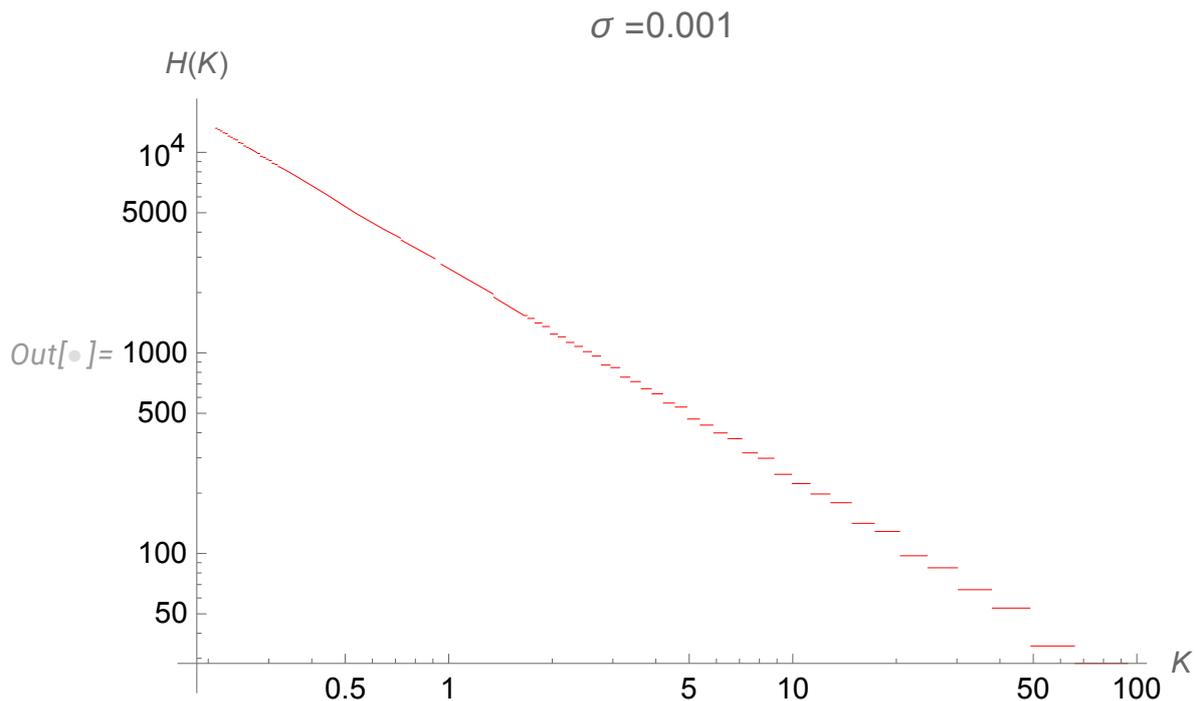
$$\sigma = 0.001$$



**Figure 10.** The log-log plot of $H(K)$ as a function of $K = |\vec{k}|\sqrt{\nu(t + t_0)}$. For display purposes, the delta function was replaced by the normal distribution PDF with $\sigma = 10^{-3}$. This change did not affect the curve, as these discrete spectrum peaks are numerically very small.

## 4. Conclusions

There are several new results reported in this paper.

- Continuum limit of distribution of scaling variables (34),(39) in the small Euler ensemble.
- Relation between the vorticity correlation and conditional probability $\mathbb{P}(Z \mid Y)$.
- Fast algorithm with $O(1)$ memory to simulate the big Euler ensemble as a Markov chain.
- The scaling law (77) for the conditional probability leads to the **discrete** energy spectrum (84) on top of a continuous background in (79).
- The decay at a given time goes only at small enough wavelengths. At a fixed wavelength, the decay stops after some critical time, inversely proportional to the wavelength.

Nothing like these quantum effects was observed in numerical simulations [9]. There could be several explanations (see discussion in the paper [10]). It is very difficult to numerically distinguish the devil's staircase like that in Figure 10 from the good old power law.

Discrete energy spectrum in classical turbulence sounds like heresy, though it does not contradict any known fluid dynamics principles. It is worth reminding that we did not assume a discrete spectrum: it came out of large scale simulations on a supercomputer with very small statistical errors and some exact distributions derived from the number theory.

We regard this discrete spectrum as a consequence of the exact analogy between decaying turbulence and the quantum mechanics in loop space [1,4,8].

We would like to hear the response from leading experts in fluid dynamics before finally publishing this paper in a journal.

**Appendix A. The cotangent sum and Jordan totients.**

The Euler ensemble expectation value of $\cot^2\left(\frac{\pi p}{q}\right)$ can be reduced to the prime numbers as follows. We start with an unconstrained sum, which is elementary [12].

$$\sum_{n=1}^{m-1} \cot^2\left(\frac{\pi n}{m}\right) = \frac{1}{3}m^2 - m + \frac{2}{3} \tag{A1}$$

Let us consider a sum $G[F, m]$ of arbitrary function $F(n/m)$ constrained to the coprime $n, m$. In our case

$$F(x) = \begin{cases} 0 \text{ if x =1} \\ \cot^2(\pi x) \text{ otherwise} \end{cases} \tag{A2}$$

Such sum satisfies the following equation (with $p_1 < p_2 \cdots < p_S$ denoting $S$ ordered prime factors of $m$ and $(n, m)$ denoting coprime $n, m$).

$$m = \prod_{s=1}^{S} p_s^{\alpha_s}; \tag{A3}$$

$$G[F, m] = \sum_{\substack{n=1 \\ (n,m)=1}}^{m} F(n/m); \tag{A4}$$

$$H(m) = \sum_{n=1}^{m} F(n/m); \tag{A5}$$

$$G[F, m] = H(m) + \sum_{s=1}^{S} (-1)^s \sum_{0<l_1<l_2\cdots<l_s\leq S} H\left(\frac{m}{p_{l_1} p_{l_2} \cdots p_{l_s}}\right); \tag{A6}$$

Let us go into detail. Consider the first term for particular $l_1 = l, p(l_1) = p$

$$H(m) - H\left(\frac{m}{p}\right) = \sum_{n=1}^{m} F(n/m) - \sum_{n'=1}^{m'} F(n'/m')\bigg|_{m'=m/p} =$$

$$\sum_{n=1}^{m} F(n/m) - \sum_{\substack{n=1 \\ n \pmod{p}=0}}^{m} F(n/m) \tag{A7}$$

We observe that the second term removes from the total sum $\sum_{n=1}^{m}$ in the first term all the terms with $n \pmod{p} = 0$. In the same way, the other terms in the sum $\sum_l H\left(\frac{m}{p_l}\right)$ remove all the terms in the first sum with $(n, m) = p_l$. However, there are terms in $\sum_{n=1}^{m} F(n/m)$ like $n = p_1 p_2$, which are proportional to two prime factors $p_1, p_2$, and we removed these terms twice, once in the term $-H\left(\frac{m}{p_1}\right)$ and the second time in $-H\left(\frac{m}{p_2}\right)$. So, we have to add them back, with +1 sign for each pair $p_i, p_j$. This addition provides the next term with double sum $\sum_{0<l_1<l_2\leq S}$.

In general, this formula is a particular case of the inclusion-exclusion principle [13]. As the basic equation (A6) is a linear functional of $H$, we can solve this equation separately for $H_l(m) = m^l$, and then by adding these solutions with proper coefficients, we get the solution for our particular

$H(m) = 2/3 - m + 1/3 m^2$. Let us start with the simplest case, $H_1(m) = m$. The solution is the Euler $\varphi(m)$. Here is how the equation is satisfied:

$$G_1(m) = m + m \sum_{s=1}^{S} (-1)^s \sum_{0 < l_1 < l_2 \cdots < l_s \leq S} \frac{1}{p_{l_1} p_{l_2} \cdots p_{l_s}} =$$

$$m \prod_{l=1}^{S} \left(1 - \frac{1}{p_l}\right) \tag{A8}$$

The next case, $G_2(m)$ is processed the same way, with the result

$$G_2(m) = m^2 \sum_{s=0}^{L} (-1)^s \sum_{l_1 < l_2 \cdots < l_s} \frac{1}{p^2(l_1) p^2(l_2) \cdots p^2(l_s)} =$$

$$m^2 \prod_{l=1}^{L} \left(1 - \frac{1}{p_l^2}\right) \tag{A9}$$

Finally, the function $G_0(m)$

$$G_0(m) = \sum_{s=0}^{S} (-1)^s \sum_{l_1 < l_2 \cdots < l_s} 1 = \sum_{s=0}^{S} (-1)^s \binom{S}{s} = (1-1)^S = 0 \tag{A10}$$

Putting all together

$$S(m) = \sum_{\substack{n=1 \\ (n,m)}}^{m-1} \cot^2\left(\frac{\pi n}{m}\right) = \frac{1}{3} J_2(m) - J_1(m); \tag{A11}$$

$$J_l(m) = m^l \prod_{p|m} \left(1 - \frac{1}{p^l}\right); \tag{A12}$$

Jordan introduced these generalized totients [14] $J_l(m)$ in 1870. The asymptotic behavior of the Jordan totient summatory functions is known:

$$\sum_{m=2}^{N} J_l(m) \to \frac{N^{l+1}}{(l+1)\zeta(l+1)} \tag{A13}$$

**Appendix B. Algorithms.**

The numerical simulation of the correlation function does not require significant computer resources. It is like a simulation of a one-dimensional Ising model with long-range forces. However, a much more efficient method is to simulate the Markov process, equivalent to the Fermionic trace of the [10].

Regarding that quantum trace, we are simulating the random histories of the changes of the Fermionic occupation numbers, taking each step off history with Markov matrix probabilities.

We devised an algorithm that does not take computer memory growing with the number $N$ of points at the fractal curve. This algorithm works for very large $N \sim 10^9$, and it can be executed on the GPU in the supercomputer cluster, expanding the statistics of the simulation by massive parallelization.

We use large values of $N = 2 * 10^8$. As for the random values of fractions $\frac{p}{q}$ with $0 < p < q < N$, we used the following Python algorithm.

We used even $N, q$ and $N_\pm = N/2$, as it was shown in [4] that these are the dominant values in the local limit $\mu \to 0$.

```python
def EulerPair(self):
    M = self.m
    while (True):
        q = 2 * np.random.randint(2,M//2)
        p = np.random.randint(1, q)
        if gcd(p,q) ==1:
            if np.random.randint(0,M) < q:
                break
            pass
        pass
    return [ p, q]
```

In other words, we generated random even $q \in (2, N)$, after which we tried random $0 < p < q$ until we got a coprime pair $(p, q)$. Statistically, at large $N$, this happens with probability $6/\pi^2 \approx 0.61$ (see [15]). On average, it takes $\pi^2/6 \approx 1.64493$ attempts to get a coprime pair.

Once we have a $p$ candidate, we accept it with the chance $w = \frac{q}{N}$, which, again, for large $q \sim N$ has a finite acceptance rate. The probabilities multiply to the desired factor, which is proportional to Euler totient

$$\mathbb{P}((p, q) = 1; p < q) = \frac{\varphi(q)}{q}; \tag{A14}$$

$$\frac{\varphi(q)}{q} \mathbb{P}(\text{random}(1, N) < q) = \frac{\varphi(q)}{N}; \tag{A15}$$

The main task is to generate a sequence of random spins $\sigma_1 = \pm 1, \ldots \sigma_N = \pm 1$ with prescribed numbers $N_+, N_-$ of $+1$ and $-1$ values. The statistical observables are additive and depend upon the partial sums $\alpha_n = \sum_{k<n} \sigma_k$.

We avoid storing arrays of $\sigma_k, \alpha_k$, using iterative methods to compute these observables. Our RAM will not grow with $N$, with CPU time growing only linearly.

The **C++** class **RandomWalker** generates this stream of $\sigma_k, \alpha_k$.

```cpp
#pragma once

#include <random>

class RandomWalker
{
public:
    RandomWalker(std::int64_t N_pos, std::int64_t N_neg) :
     N_pos(N_pos), N_neg(N_neg), alpha(), gen(std::random_device{}()), unif(0, 1) {

    }

    int Advance()
    {
        int sigma = RandomSign();
        (sigma == 1 ? N_pos : N_neg)--;
        alpha += sigma;
        return sigma;
    }

    std::int64_t get_alpha() const { return alpha; }

private:
    int RandomSign()
```

```
25        {
26            return (unif(gen) * double(N_pos + N_neg) <= N_neg) ? -1 : 1;
27        }
28
29        std::int64_t N_pos;
30        std::int64_t N_neg;
31        std::int64_t alpha; // alpha[i]
32        std::minstd_rand gen;
33        std::uniform_real_distribution<double> unif;
34    };
```

At each consecutive step, the random sign $\sigma = \pm 1$ is generated with probabilities $\frac{n_+}{n_+ + n_-}, \frac{n_-}{n_+ + n_-}$, after which the corresponding number $n_+$ or $n_-$ is decremented, starting with $n_\pm = N_\pm$ at step zero. By construction, in the end, there will be precisely $N_+$ positive and $N_-$ negative random values $\sigma_k$.

This algorithm describes a Markov chain corresponding to sequential drawing random $\sigma$ from the set with initial $N_\pm$ positive and negative spins, and equivalent to one-dimensional random walk on a number line. These conditional probabilities

$$\mathbb{P}\left(\sigma_k = 1 | \sigma_1 \ldots \sigma_{k-1}\right) = \frac{n_+}{n_+ + n_-}; \tag{A16}$$

$$n_\pm = \sum_{l<k} \frac{1 \pm \sigma_l}{2} \tag{A17}$$

are such that unconditional probability equals to the correct value:

$$\left\langle \mathbb{P}\left(\sigma_k = 1 | \sigma_1 \ldots \sigma_{k-1}\right)\right\rangle_{\sigma_1 \ldots \sigma_{k-1}} = \mathbb{P}\left(\sigma_k = 1\right) = \frac{N_+}{N_+ + N_-}; \tag{A18}$$

regardless of the position $k$ of the variable $\sigma_k$ at the chain.

The formal proof goes as follows.

**Proof.** At $k = 1$, without spins to the left on the chain, the conditional probability equals the total probability, and the statement holds. With some number of spins on the left, averaging over all these spins at a fixed value of $\sigma_k$ is equivalent to averaging over all values except $\sigma_k$ because the only condition on random spins is their net sum, which is a symmetric function of their values. Thus, the average of conditional probabilities equals the total probability (A18) for any spin on the chain, not just the first one. $\square$

Here is the code, which computes the sample of $|d\vec{S}_{nm}|, \vec{\omega} \cdot \vec{\omega}$ given $n, m, N_+, N_-, \beta$. The 2D vectors $\vec{S}$ are treated as complex numbers, using 128 bit complex arithmetic.

```
1    #include <complex>
2    #include <cassert>
3    #include <iostream>
4    #include "Euler.h"
5    #include "RandomWalker.h"
6    #include "MatrixMaker.h"
7
8    using complex = std::complex<double>;
9    using namespace std::complex_literals;
10
11   inline std::complex<double> expi(double a)
12   {
13       double sin_a, cos_a;
14       sincos(a, &sin_a, &cos_a);
15       return {cos_a, sin_a};
```

```
16   }
17
18   double DS(std::int64_t n, std::int64_t m,
19   std::int64_t N_pos, std::int64_t N_neg, double beta, /*OUT*/ double *o_o)
20   {
21       assert(n < m);
22       std::int64_t M = N_pos + N_neg;
23       int sigma_n, sigma_m, alpha_m, alpha_n;
24       complex S_nm, S_mn;
25       double R_nm;
26
27       RandomWalker walker(N_pos, N_neg);
28       std::int64_t i = 0;
29       for (; i != n; i++)
30       { // i = [0; n)
31           S_mn += expi(walker.get_alpha() * beta);
32           walker.Advance();
33       }
34
35       alpha_n = walker.get_alpha();
36       S_nm += expi(alpha_n * beta);
37       sigma_n = walker.Advance(); // i = n
38       for (i++; i != m; i++)
39       { // i = (n, m)
40           S_nm += expi(walker.get_alpha() * beta);
41           walker.Advance();
42       }
43
44       alpha_m = walker.get_alpha();
45       S_mn += expi(alpha_m * beta);
46       sigma_m = walker.Advance(); // i = m
47       for (i++; i != M; i++)
48       { // i = (m, M)
49           S_mn += expi(walker.get_alpha() * beta);
50           walker.Advance();
51       }
52
53       *o_o = -M * (M - 1) / 2 * sigma_n * sigma_m / (2 * pow(tan(beta / 2), 2)) \
54           * pow(sin(beta / 4 * (2 * (alpha_m - alpha_n) + sigma_m - sigma_n)), 2);
55
56       S_nm /= double(m - n);
57       S_mn /= double(n + M - m);
58       return abs((S_nm - S_mn) / (2 * sin(beta / 2)));
59   }
```

This code fits the architecture of the supercomputer cluster with many GPU units at each node. These GPU units are aggregated in blocks by 32, with every init inside the block working in sync. We use this synchronization to collect statistics: every block is assigned with different random values of $n, m, N_+, N_-, \beta$, and every unit inside the block performs the same computation of **DS** function with different values of the integer seed for the random number generator, automatically provided by the system.

When quantum computers become a reality, our sum over the sum of the Markov history of discrete spin (or Fermion) variables will run massively parallel on the qubits. Our Euler/Markov ensemble is ideally suited for quantum computers.

**Appendix C. The $O(3)$ group average**

The correlation function (60) involves an integral over the $O(3)$ group

$$Q(\vec{r}, \vec{s}) = \int_{O(3)} \frac{d\Omega}{|O(3)|} \exp\left(\imath \vec{r} \cdot \hat{O} \cdot \vec{s}\right); \tag{A19}$$

with the vector $\vec{s}$ related to our fractal curve $\vec{F}(\theta)$ in a particular sample in our ensemble.

Both vectors $\vec{r}, \vec{s}$ are real.

This integral is a particular case of a Harish-Chandra-Itzykson-Zuber integral formula [16],

$$\int_{U(n)} e^{\text{tr}(AUBU^*)} dU = \left(\prod_{p=1}^{n-1} p!\right) \frac{\det\left[e^{a_i b_j}\right]}{\Delta(A)\Delta(B)} \tag{A20}$$

where

$$\Delta(A) = \prod_{i<j}(a_j - a_i) \tag{A21}$$

is the Vandermonde determinant.

In our case, $O(3) = \left(SU(2)\big/\mathbb{Z}_2\right)$, so we use $n = 2$ formula with

$$A = \imath \sigma_i r_i, \ \Delta(A) = 2\imath |\vec{r}| \tag{A22}$$

$$B = \sigma_j s_j, \ \Delta(B) = 2|\vec{s}| \tag{A23}$$

$$\det\left[e^{a_i b_j}\right] = 2\imath \sin\left(|\vec{r}||\vec{s}|\right) \tag{A24}$$

where $\vec{\sigma}$ are Pauli matrices. With proper normalization to 1 at $\vec{r} = 0$, the HCIZ integral reads

$$\int_{O(3)} \frac{d\Omega}{|O(3)|} \exp\left(\imath \vec{r} \cdot \hat{O} \cdot \vec{s}\right) = \frac{\sin\left(|\vec{r}||\vec{s}|\right)}{|\vec{r}||\vec{s}|} \tag{A25}$$

Next, let us consider the case of the Fourier integral needed for the Wilson loop.

The group integration is now involved in the averaging of the ensemble of our random loops $\vec{F}_k$ with vector steps $\vec{q}_k = \vec{F}_{k+1} - \vec{F}_k$:

$$\int_{O(3)} \frac{d\Omega}{|O(3)|} \exp\left(-\frac{\imath \sum_{k=1}^{N} \vec{C}(2\pi k/N) \cdot \hat{\Omega} \cdot \vec{q}_k}{\sqrt{2\nu(t + t_0)}}\right) =$$

$$\int_{O(3)} \frac{d\Omega}{|O(3)|} \exp\left(\imath \text{tr}\,\hat{\Omega} \cdot \hat{X}\right); \tag{A26}$$

$$\hat{X} = -\frac{\sum_{k=1}^{N} \vec{q}_k \otimes \vec{C}(2\pi k/N)}{\sqrt{2\nu(t + t_0)}} \tag{A27}$$

The matrix $\hat{X}$ is not symmetric but real, as the vectors $\vec{q}_k$ are real. These are unit vectors on a plane, equivalent to complex numbers

$$\hat{q}_k = q_{k,x} + \imath q_{k,y} = \imath \sigma_k \exp\left(\imath \frac{\beta \sigma_k}{2}\right) \prod_{l=0}^{k-1} \exp\left(\imath \beta \sigma_l\right); \tag{A28}$$

This integral does not reduce to the HCIZ integral and needs special treatment.

We use a quaternionic representation of $O(3)$ and write this integral:

$$Q(\hat{X}) = \int_{S_3} \frac{(dx)}{|S_3|} \exp\left(\imath x_\lambda x_\rho R_{\lambda\rho}\right); \tag{A29}$$

$$R_{\lambda\rho} = \tfrac{1}{2}\left(T_{ij}^{\lambda\rho} + T_{ij}^{\rho\lambda}\right) X_{ij}; \tag{A30}$$

$$T_{ij}^{\alpha\beta} = \tfrac{1}{2}\mathrm{tr}\, \sigma_i \tau_\alpha \sigma_j \tau_\beta^\dagger; \tag{A31}$$

$$\tau_\alpha = \{1, \imath\vec{\sigma}\}; \ \alpha = (0,1,2,3); \tag{A32}$$

$$\sum x_\alpha^2 = 1 \tag{A33}$$

In this representation, we have full $O(4)$ symmetry of this integral over unit sphere $S_3$ in four dimensions, plus there is a symmetry of the tensor $R_{\alpha\beta} = R_{\beta\alpha}$.

This Fourier integral can be computed in tens of milliseconds using a fast Python library **quadpy**[17]. It is optimized for integration over geometric shapes, including spheres in arbitrary dimensions. We licensed this library from the author and used it for our computations. Here is the code with its output for a random real symmetric tensor $R$

```python
def SphericalFourierIntegral(R):
    dim = 4
    scheme = quadpy.un.mysovskikh_2(dim)
    print("tol=", scheme.test_tolerance)
    vol = scheme.integrate(lambda x: np.ones(x.shape[1]), np.zeros(dim), 1.0)

    def func(x):
        return np.exp(1j * np.sum(x * (R.dot(x)), axis=0))/ vol
    return scheme.integrate(func, np.zeros(dim), 1.0)
    '''
    QuadPy.py::test_Fourie03Integral PASSED                          [100%]tol= 1e-14

    quadpy: SphericalFourierIntegral = (-0.05635295876166728-0.012727710917488425j)
    quadpy O(3) Fourier Integral 28.73 ms


    '''
```

## References

1. Migdal, A. Loop Equation and Area Law in Turbulence. In *Quantum Field Theory and String Theory*; Baulieu, L.; Dotsenko, V.; Kazakov, V.; Windey, P., Eds.; Springer US, 1995; pp. 193–231. https://doi.org/10.1007/978-1-4615-1819-8.
2. Makeenko, Y.; Migdal, A. Exact equation for the loop average in multicolor QCD. *Physics Letters B* **1979**, *88*, 135–137. https://doi.org/10.1016/0370-2693(79)90131-X.
3. Migdal, A. Loop equations and $\frac{1}{N}$ expansion. *Physics Reports* **1983**, *201*.
4. Migdal, A. To the Theory of Decaying Turbulence. *Fractal and Fractional* **2023**, *7*, 754, [arXiv:physics.flu-dyn/2304.13719]. https://doi.org/10.3390/fractalfract7100754.
5. Hardy, G.H.; Wright, E.M. *An introduction to the theory of numbers*, sixth ed.; Oxford University Press, Oxford, 2008; pp. xxii+621. Revised by D. R. Heath-Brown and J. H. Silverman, With a foreword by Andrew Wiles.
6. Basak, D.; Zaharescu, A. Connections between Number Theory and the theory of Turbulence, 2024. To be published.
7. Migdal, A. "NumericalAnalysisOfEulerEnsembleInDecayingTurbulence". "https://www.wolframcloud.com/obj/sasha.migdal/ Published/NumericalAnalysisOfEulerEnsembleInDecayingTurbulence.nb", 2023.
8. Migdal, A. Statistical Equilibrium of Circulating Fluids. *Physics Reports* **2023**, *1011C*, 1–117, [arXiv:physics.flu-dyn/2209.12312]. https://doi.org/10.48550/ARXIV.2209.12312.
9. Sreenivasan, K.R. On the scaling of the turbulence energy dissipation rate. *The Physics of Fluids* **1984**, *27*, 1048–1051, [https://aip.scitation.org/doi/pdf/10.1063/1.864731]. https://doi.org/10.1063/1.864731.
10. Migdal, A. Dual Theory of Decaying Turbulence: 1: Fermionic Representation., 2023.

11. Migdal, A. LoopEquations. https://github.com/sashamigdal/LoopEquations.git, 2023.

12. Migdal, A. Decaying Turbulence Computations. https://www.wolframcloud.com/obj/sasha.migdal/Published/DecayingTurbulenceComputations.nb, 2023.

13. Cioabă, S.M.; Murty, M.R., The Principle of Inclusion and Exclusion. In *A First Course in Graph Theory and Combinatorics: Second Edition*; Springer Nature Singapore: Singapore, 2022; pp. 33–42. https://doi.org/10.1007/978-981-19-0957-3_3.

14. Moree, P.; Saad Eddin, S.; Sedunova, A.; Suzuki, Y. Jordan totient quotients. *Journal of Number Theory* **2020**, *209*, 147–166. https://doi.org/10.1016/j.jnt.2019.08.014.

15. Lei, J.; Kadane, J.B. On the probability that two random integers are coprime, 2019, [arXiv:math.PR/1806.00053].

16. McSwiggen, C. The Harish-Chandra integral: An introduction with examples, 2021, [arXiv:math-ph/1806.11155].

17. Schlömer, N. QuadPy. https://github.com/sigma-py/quadpy.git, 2023.