The Koch fractal drum and its vibrational mechanics

Candidate ID number: 10058

ABSTRACT

This study investigates the vibrational properties of a square Koch fractal drum using numerical methods to solve the Helmholtz equation. By implementing finite difference methods with both five-point and nine-point stencils, the eigenfrequencies and vibrational modes of the fractal drum were calculated. The results confirm the existence of localized vibrational modes, which differ from those observed in regular, non-fractal drums. Additionally, the eigenfrequencies obtained align with prior theoretical predictions, demonstrating that fractal boundaries significantly influence wave behavior. The Weyl-Berry conjecture was tested, yielding a fractal dimension consistent with theoretical expectations for low fractal levels. Computational limitations restricted analysis to relatively low fractal levels, highlighting the need for effective programming schemes, both for run time and memory usage. These findings contribute to the broader question of whether one can "hear the shape of a drum."

Keywords: Square Koch fractal, Finite difference schemes, Weyl-Berry conjecture.

I. INTRODUCTION

A. General Introduction

A regular drum will vibrate with an eigen-frequency when hit with a drumstick. Given its (often) round shape, the entire drum surface will vibrate in the same mode. However, can a drum with localized vibrating modes exist? That is what this paper will investigate. By introducing a square Koch fractal into the surface of the drum, Mark Kac explored whether one can "hear the shape of the drum."[1] Will changing the perimeter of the drum from a circle to a fractal alter the vibrating modes and eigenfrequencies?

To investigate this question, this report examines the vibrational modes of a square Koch fractal drum, using numerical techniques to solve the Helmholtz equation. The use of a fractal boundary introduces complexity due to the self-similar and non-smooth nature of the perimeter. Fractals, such as the square Koch curve, are mathematical objects that exhibit self-similarity at different scales and are used in many physical and natural systems, including coastlines, biological structures, and wave propagation in irregular media. Their non-integer dimensions make them particularly interesting in physical contexts where geometry significantly affects dynamics, such as wave behavior.

By exploring how waves behave inside fractal structures, this study contributes to the understanding of wave mechanics in complex geometries.

B. Structure and Organization

This paper will focus on:

Theory in section II gives the mathematical formulation of the problem. The creation of the fractal drum will be presented, and the mathematical formulation of the problem.

Models and methods in section III presents the mathematical models and numerical methods used to analyze the vibrational properties of the Koch fractal drum. This includes a discussion on the classification of points within the fractal geometry and the finite difference schemes employed to solve the eigenvalue problem.

Results and discussion in section IV details the results obtained from numerical simulations, focusing on the vibrational modes, eigenfrequencies, and the implications of fractal boundaries on wave behavior. A comparison between five-point and nine-point stencil methods is also provided. Additionally, this section explores the Weyl-Berry conjecture and evaluates its applicability to the Koch fractal drum.

Conclusions in section V summarizes the key findings, discussing their significance in the context of wave mechanics and fractal geometries. Finally, potential improvements and directions for future research are suggested, particularly in addressing computational challenges and extending the analysis to higher fractal levels.

II. THEORY

A. The Vibrating Fractal Drum

To study this in theory, one approach is to numerically solve the Helmholtz equation:

$$-\nabla^2 U(\mathbf{r},\omega) = \frac{\omega^2}{c^2} U(\mathbf{r},\omega), \quad \text{in } S \tag{1a}$$

$$U(\mathbf{r},\omega) = 0, \quad \text{on } \partial S \tag{1b}$$

Where $U(\mathbf{r},\omega)$ is the eigen-mode of the wave corresponding to the angular eigen-frequency ω and point $\mathbf{r}$, c is a velocity and thus ω^2/c^2 is an eigenvalue of U, S is the surface of the fractal drum and ∂S is the boundary of the fractal drum. This equation can be derived by Fourier transforming the wave equation $\nabla^2 u = (1/v^2)\partial_t^2 u$. To solve equation (1), two different finite difference stencils will be used. A five-point stencil and a 9-point stencil. By applying these[2], the five-point stencil

becomes

$$\nabla^2 U_{0,0} = \left(\frac{\partial^2 U}{\partial x^2} + \frac{\partial^2 U}{\partial y^2}\right) 0,0 = \frac{\omega^2}{c^2} U 0,0 \\ = \frac{1}{h^2}\left(U_{1,0} + U_{0,1} + U_{-1,0} + U_{0,-1} - 4U_{0,0}\right) + O(h^2) \tag{2}$$

and the nine-point stencil becomes

$$\nabla^2 U_{0,0} = \left(\frac{\partial^2 U}{\partial x^2} + \frac{\partial^2 U}{\partial y^2}\right) 0,0 = \frac{\omega^2}{c^2} U 0,0 \\ = \frac{1}{12h^2}[-60U_{0,0} + 16(U_{1,0} + U_{0,1} + U_{-1,0} + U_{0,-1}) \\ - (U_{2,0} + U_{0,2} + U_{-2,0} + U_{0,-2})] + O(h^4). \tag{3}$$

Where $U_{0,0}$ is the initial point, $U_{\pm 1,0}$ and $U_{\pm 2,0}$ are the nearest and next-nearest neighbors in the horizontal direction respectively and $U_{0,\pm 1}$ and $U_{0,\pm 2}$ are the nearest and next-nearest neighbors in the vertical direction respectively. h is the step between each point, i.e. the lattice constant, which will be defined later.

To investigate if a drum can be heard, equation (1) will be solved on the drum surface, S, in the shape of a Koch fractal of level ℓ should give the answer if it is possible to hear the shape of exactly this drum.

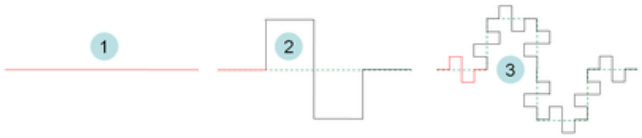

FIG. 1: The first steps for creating a square Koch fractal, for $l = 0, 1, 2$. The figure is taken from Wikipedia[3].

To create the fractal, one starts with a single line, splits it into four equal parts, and creates the shape shown in figure 1. By creating four such segments and putting them together, one gets the result shown in figure 2.

In addition, this is not far from the problem of the clamped thin plate, with equations

$$\nabla^4 W(\mathbf{r}, \omega) = \lambda W(\mathbf{r}, \omega), \quad \text{in } S \tag{4a}$$

$$W(\mathbf{r}, \omega) = 0, \quad \text{on } \partial S \tag{4b}$$

$$(\hat{n} \cdot \nabla) W(\mathbf{r}, \omega) = 0, \quad \text{on } \partial S. \tag{4c}$$

Where $\hat{n}$ is the outward normal vector of S and ∇^4 is the biharmonic operator with discretization:

$$\nabla^4 U_{0,0} = \left(\frac{\partial^4 U}{\partial x^4} + 2\frac{\partial^4 U}{\partial x^2 \partial y^2} + \frac{\partial^4 U}{\partial y^4}\right) 0,0 \\ = \frac{1}{h^4}[20U0,0 - 8(U_{1,0} + U_{0,1} + U_{-1,0} + U_{0,-1}) \\ + 2(U_{1,1} + U_{1,-1} + U_{-1,1} + U_{-1,-1}) \\ + (U_{2,0} + U_{0,2} + U_{-2,0} + U_{0,-2})] + O(h^2)^2 \tag{5}$$

A similar eigenvalue problem to the fractal drum.

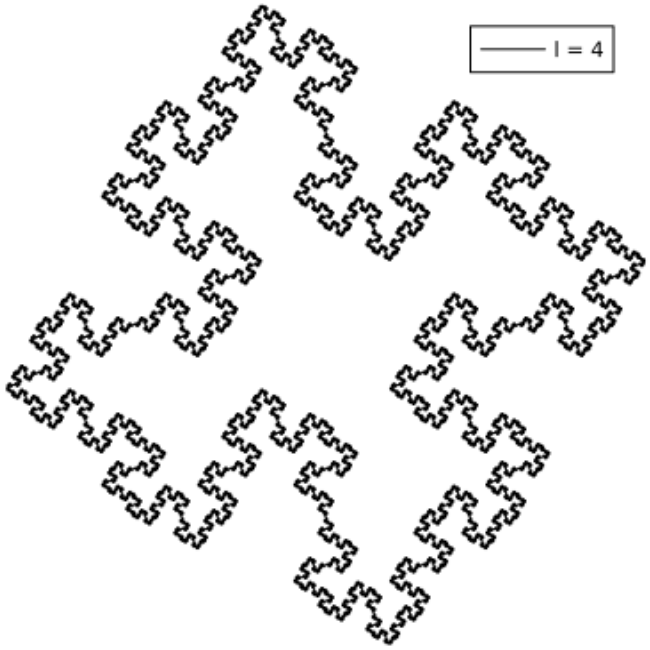

FIG. 2: A square Koch fractal drum surface, with level $\ell = 4$. One can see that it can be divided into four separate and equal quadrants which all meet in the middle.

III. MODELS AND METHODS

The entire programming part of this project was done in Julia.

As was stated in section II A, finite differences have been used in this paper to solve eigenvalue problems, on the form $A\mathbf{v} = \lambda\mathbf{v}$, where A is the coefficient matrix corresponding to the finite difference approximation of U, $\mathbf{v}$ is an eigenvector and λ the corresponding eigenvalue. By multiplying both sides of equations (2) and (3) by L^2 and define the dimensionless eigenfrequency

$$\Omega = \frac{\omega}{c} L. \tag{6}$$

Then Ω is the eigenvalue of the fractal drum to be solved for.

A. Classification of points

To solve the system, it is vital to know the geometry of the drum. Specifically, which points in the square grid are outside of, inside of or on the lattice. The points were given the values -1 if they were outside, 1 if they were inside and 0 if they were on the fractal. Now there are several ways to do this. Two different methods were chosen for this project, a winding number algorithm created from scratch and the LibGEOS package in Julia[4].

The winding number algorithm takes in one point $p = (x, y)$ of the square grid at a time. Then it loops over all vertices $\{q_j, q_{j+1}\}$ in the fractal, where $j \in \mathbb{N}_1^{M-1}$, where M is the number of points on the fractal (the number of vertices).

Given a set of vertices defining a closed polygon, the winding number at a point p is computed as follows:

1. Initialize the winding number $WN = 0$.
2. Iterate through each vertex of the fractal:
   - If the edge crosses the horizontal line extending from (x, y) in an upward direction, check whether the point lies to the left of the edge. If so, increment WN.
   - If the edge crosses downward, check whether the point lies to the right of the edge. If so, decrement WN.
3. Return the final value of WN.

Then if $WN \neq 0$ it lies inside the fractal and if $WN = 0$ it lies outside. This method does not check whether or not a point lies on the fractal, but this was easily done by checking if a certain coordinate lies in the fractal or not. This method used ~ 12 seconds for $\ell = 4$ and was not tested for $\ell = 5$ as it was the (barely) slower option.

The LibGEOS package has a function called "within" which takes in a a LibGEOS.polygon (the fractal) and a LibGEOS.point (a given point p). It then returns 1 if the point is inside the polygon, 0 if it is outside the polygon and -1 if it is on the polygon. This method used ~ 10 seconds for $\ell = 4$ and ~ 50 minutes for $\ell = 5$.

The two huge drawbacks for both methods, is the looping through square grid. To classify all points in the square grid, one must loop over all points in the square grid, only for then to loop over all points on the fractal. As shown in table **??**, this quickly reaches into the order of millions, giving rise to long computation times. Which was the main reason as to why the program was not run for any $\ell > 5$.

B. Solving the system

To map the eigenvector $\mathbf{v}$ to internal lattice points $U_{m_i n_i}$, the convention $\mathbf{v} = (U_{m_1 n_1}, U_{m_2 n_2}, ...)$ was adopted. Where $m_i n_i$ is a set of indices defined by a classification matrix C, such that $C(m_i, n_i) = i$ where $i \in \mathbb{N}$[5].

To solve the system, a sparse matrix A initialized such that all elements were set to 0. The program iterated over all points in the square grid column by column. If $i = C(m_i, n_i) = 0$, the program did nothing. If $i = C(m,n) > 0$, then $A_{ii} = 4/h^2 = 4/d_\ell^2 = 4^{2\ell+1}$. Then the program checked the neighbors $j = C(m+1, n)$, $j = C(m-1, n)$, $j = C(m, n+1)$ and $j = C(m, n-1)$. For each $j > 0$ then $A_{ij} = -1/h^2 = -4^{2\ell}$ for the five-point stencil. For the nine-point stencil, both nearest and next nearest neighbors were checked with corresponding values $A_{ij} = 4^{2\ell+1}/3$ and $A_{ij} = -4^{2\ell}/12$ for $j > 0$.

The same method was used for equation (4), with its coefficients: $20/h^4$ for the diagonal, $-8/h^2$ for nearest neighbors, $2/h^4$ for nearest diagonal neighbors and $1/h^4$ for next nearest neighbors. The extra boundary condition in the equation means that the rate of change on the boundary must be 0. By discretizing $\hat{n} \cdot \nabla$, one gets that for this to be the case, the points inside the fractal with a neighbor on the boundary must be 0 at all times. Effectively, this is just moving the boundary of the fractal one step inwards, and then using the Dirichlet boundary condition on the new boundary. For solving the eigenvalue problem, the function "eigs(...)" from the package "LinearAlgebra" was used.

An approximation for the size of matrix A is $\mu_\ell \times \mu_\ell$, where $\mu_\ell \approx \left\lfloor (N_\ell + 1)^2 L^2 / L_\ell^2 \right\rceil$, which as shown above will become a large number, even for small ℓ. For $\ell = 5$ and $L = 1$, $\mu_\ell^2 \approx 8.08 \cdot 10^6$, where most elements are 0. Which is why A was initialized to be a sparse matrix, to reduce the use of memory. If A was initialized to a regular matrix, it would demand ~ 2.8 times more memory than the square grid.

In this project, the same system was initialized every time, so there is no need to set seeds to be able to reproduce results.

IV. RESULTS AND DISCUSSION

A. Vibrating modes

As can be seen in figure 3, one can clearly see localized vibrating modes in different quadrants of the fractal drum. This would not be the case for a regular square drum. This is the case for several of the higher order modes as well. We can also see degeneracy in the first excited mode, as predicted by the data in table I.

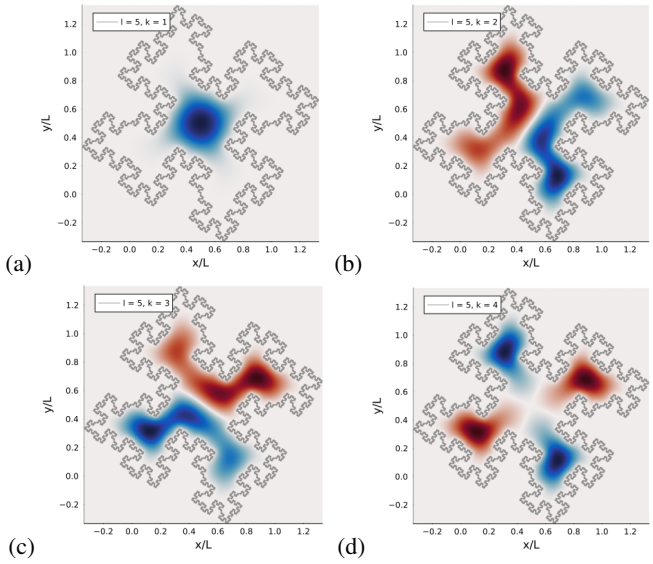

FIG. 3: The first four vibrating modes for $\ell = 5$ using the five-point stencil. (a) is $k = 1$, (b) is $k = 2$, (c) is $k = 3$ and (d) is $k = 4$.

When presenting the results below, there has been added a fourth column including the dimensionless eigenfrequency Ω_k divided by $\Omega^* = \sqrt{2}\pi$. This is the fundamental eigenfrequency of a regular square drum[5]. According to Sapoval *et.*

al, the fundamental eigenfrequency of the fractal drum should be $\Omega_0 = 2.100\Omega^*$[6]. In tables I and II, we see that this program's estimate is $\sim 1\%$ larger for both stencils. Which is a good estimate for such a low fractal level $\ell = 5$. We see that the difference between the two stencils is less than 1%, with the five-point stencil actually being a bit closer to Sapoval *et. al*'s estimate.

k	g_k	Ω_k	Ω_k/Ω^*
0	1	9.42901	2.12227
1	2	14.1445	3.18362
1	2	14.1445	3.18362
3	1	14.4174	3.24505
4	1	14.4946	3.26242
5	2	15.0818	3.39459
5	2	15.0818	3.39459
7	1	17.6548	3.97372
8	1	18.9099	4.25622
9	1	19.4556	4.37905
10	1	20.0218	4.50649
11	1	20.5984	4.63626
12	1	21.3460	4.80453
13	1	21.6370	4.87003
14	1	23.3236	5.24966
15	1	23.5829	5.30801

TABLE I: Eigenfrequencies for the 5-point stencil for $\ell = 5$. Where g_k is the degeneracy for mode number k.

k	g_k	Ω_k	Ω_k/Ω^*
0	1	9.43287	2.12314
1	2	14.1535	3.18566
1	2	14.1535	3.18566
3	1	14.4270	3.24721
4	1	14.5039	3.26452
5	2	15.0883	3.39605
5	2	15.0883	3.39605
7	1	17.6632	3.97562
8	1	18.9199	4.25847
9	1	19.4660	4.38138

TABLE II: Eigenfrequencies for the 9-point stencil for $\ell = 5$.

In figure 4, one can see the vibrating modes for two relatively high frequencies. Here, one can clearly see how the fractal perimeter of the drum creates a complex swinging mode. One can see there are tens of distinct amplitudes with different sizes throughout the surface. They are also symmetric, and arranged in such a way that every other amplitude is of the same sign. This, almost chaotic, behavior is far from what one would see for any swinging mode on any normal drum surface (with a normal perimeter of course).

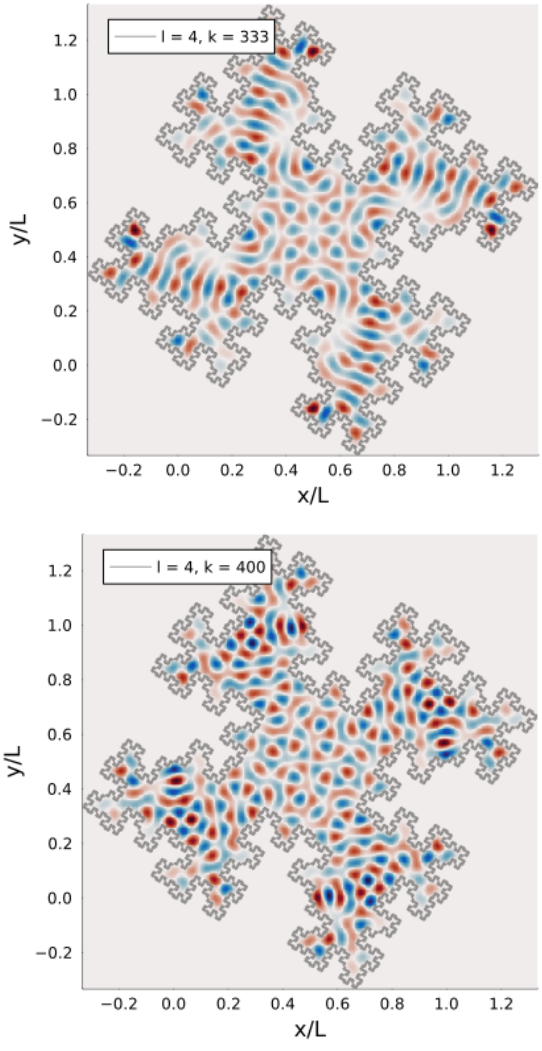

FIG. 4: The vibrating modes $k = 333$ (a) and $k = 400$ for (b), both are solved using the five-point stencil for l=4.

B. Clamped thin plate

When solving for the clamped thin plate, the program crashed when trying to calculate eigenvalues for $\ell = 5$, even when using 10^5 max iterations and a tolerance of 10^{-3}, and calculating only the lowest two eigenvalues. Which is why table III shows values for $\ell = 4$.

k	g_k	Ω_k	Ω_k/Ω^*
0	1	30686.2	6906.82
1	1	125629	28276.5
2	1	206381	46451.9
3	1	257040	57854.2

TABLE III: Eigenfrequencies for the biharmonic eigenvalue problem for $\ell = 4$.

Even then, one can see that the eigenfrequencies of the

clamped thin plate are $3-4$ orders of magnitude larger than the eigenfrequencies for the fractal drum. This is the result of limiting the area of the fractal drum to a smaller one compared to the original fractal drum, together with a different finite difference scheme.

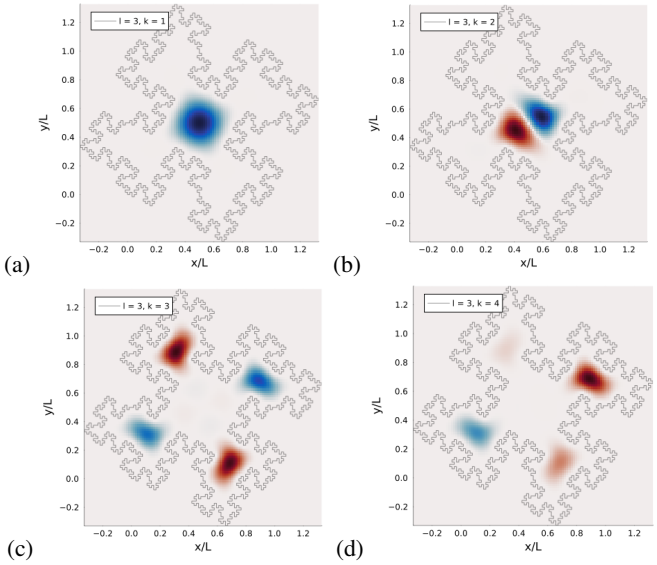

FIG. 5: The first four vibrating modes for $\ell = 3$ for the thin clamped plate. (a) is $k = 1$, (b) is $k = 2$, (c) is $k = 3$ and (d) is $k = 4$.

In figure 5, one can see the 4 lowest order vibrating modes for the thin clamped plate for $\ell = 4$. Comparing to the fractal drum modes, one can see that the modes for the thin clamped plate have amplitudes with lower area. The first mode is quite similar to the fractal drum, but then they diverge. There are similarities between the two, in that both have a low number of amplitudes in its surface. Other than that, one can see that the amplitudes for the second, third and fourth modes are located either just in the middle or just in the four arms, and not in both. While the second and third modes are anti-symmetric, the fourth mode is not, why that is not the case is not clear. All higher order modes which were retrieved and plotted were anti-symmetric. Since the algorithm was run on a high max iterations and high convergence threshold, which might cause some slight inaccuracies.

C. Weyl-Berry conjecture

The linear regression of $\log(\Delta N(\Omega_k))$ and $\log(\Omega_k)$ was done using 400 datapoints for $\ell = 4$. The fractal dimension was $D = 1.444$ with a standard deviation of 0.034, giving $D \in [1.377, 1.511]$ with the five-point stencil and with an R-squared value of $R^2 = 0.803$ for the five-point stencil. For the nine-point stencil the fractal dimension was $D = 1.454$ with a standard deviation of 0.034, giving $D \in [1.385, 1.513]$ and with an R-squared value of $R^2 = 0.794$. Both are inclusive of the expected value 1.500, with reasonably high R-squared

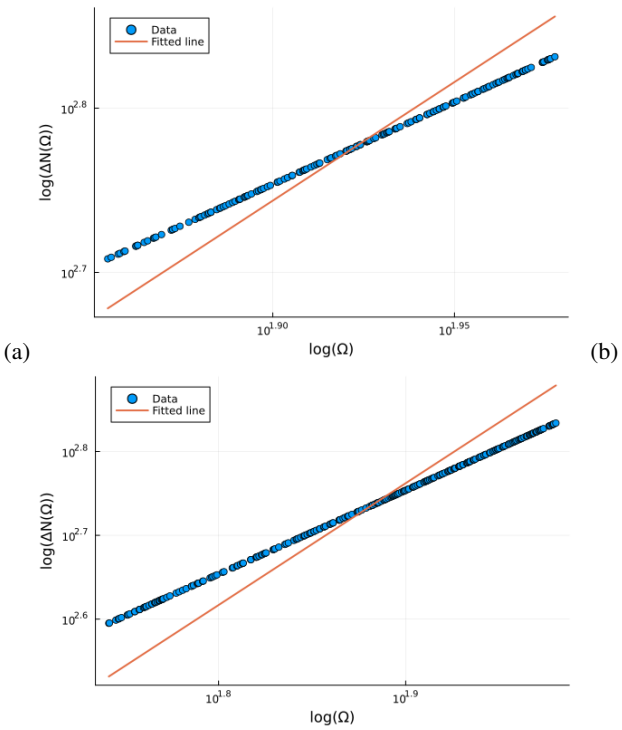

FIG. 6: Linear regression of the Weyl-Berry conjecture, using data from the five-point (a) and nine-point (b) stencils for $\ell = 4$.

values, although one would hope for higher. Also, both miss the target with a few percent, which is not desirable.

It is worth mentioning that for $\ell = 5$ for the five-point stencil, I got $D \in [1.404, 2.516]$ with $R^2 = 0.072$. Both are performing quite bad, but there is no evidence of correlation for $\ell = 5$. The data is similar for the nine-point stencil. Since the performance was quite good for $\ell = 4$, linear regression was not done for $\ell = 3$ for more than 40 values, as the results for low frequencies align well with $\ell = 4$.

Since the Weyl-Berry conjecture holds in the limit of large ω, only eigenfrequencies for $k > 99$ were used to find the fractal dimension. I.e. the fractal dimension was calculated for the 400 eigenfrequencies in the interval $[100, 500]$. When choosing all frequencies $k \in [100, 500]$, the fit became quite bad and the result was $D = 1$, which is far off the true value. The value 100 was chosen such that one would still have quite a large number of datapoints, and the curve looked approximately linear after this point when looking at a scatter plot (as can also be seen in figure 6).

V. CONCLUSIONS

Together with the heightened eigenfrequencies of the fractal drum and the localized vibrating modes shown in figure 3, one can clearly see that it is possible to hear the shape of a drum. Specifically, one can clearly hear the difference from a square drum and a square Koch fractal drum. The numeri-

cal approaches in this paper successfully captures the features of the fractal drum's behavior, with results aligning well with expectations.

However, computational challenges limited the study to a low fractal level. With both memory shortages and processing time for higher levels being large hindrances for analysis of these higher levels. Further investigations would be creating more effective code to restrict the memory usage and also improve processing time used to classify points within, outside, or on the fractal geometry. This study shows the impact of fractal geometry on vibrational mechanics.

VI. ACKNOWLEDGMENT

The author of this paper thanks candidate numbers 10008 and 10033 for collaborating on this project.

REFERENCES

[1] M. Kac, "Can one hear the shape of a drum?" The American Mathematical Monthly **73**, 1–23 (1966).
[2] M. Abramowitz and I. A. Stegun, *Handbook of Mathematical Functions with Formulas, Graphs, and Mathematical Tables* (U.S. Department of Commerce, National Bureau of Standards, Washington, D.C., 1964) sections 25.3.30-32.
[3] Wikipedia contributors, "Minkowski sausage — Wikipedia, the free encyclopedia," (2025), [Online; accessed 11-Mar-2025].
[4] JuliaGeo Community, "LibGEOS.jl: Julia wrapper for the GEOS library," (2025), accessed: March 12, 2025.
[5] V. P. Simonsen, N. Hale, and I. Simonsen, "Eigenmodes of fractal drums: A numerical student experiment," (2023), arXiv:2309.13613 [physics.ed-ph].
[6] B. Sapoval, T. Gobron, and A. Margolina, "Vibrations of fractal drums," Phys. Rev. Lett. **67**, 2974–2977 (1991).