Notation

$N :=$ number of points sampled

$N \sim h^{-2}$

$B :=$ number of basis functions used in vergini method

$k :=$ wavenumber

$E = k^2 :=$ energy of eigenfunction

$h :=$ orthogonal spacing between sampled points

$\alpha := kh$

$\Omega \in \mathbb{R}^2$ compact domain

$\Gamma = \partial\Omega$ domain boundary

Chapter 1

Introduction

1.1 Motivation

Nodal domains characterize regions of a vibrational surface that move together. The boundaries between nodal domains, known as nodal line are the regions which do not vibrate at all. Understanding the characteristics of nodal domains has potential applications in musical instruments, mechanical engineering, geophysics, astrophysics, and many other areas dealing with wave behavior [3].

The nodal domains studied herein are formulated in terms of quantum mechanical wave functions on Euclidean billiards, a canonical example of quantum chaos. Quantum chaos lies at the intersection of quantum mechanics and chaos theory. The primary signature of chaos in classical systems is a nonlinear (exponential) divergence of paths in phase space of a system. Quantum mechanics however, is entirely linear and quantum chaos deals with energy eigenfunctions of systems, which are constant in time. Thus chaos in quantum systems is manifest in different ways, one of the most studied being wavefunctions in chaotic domains. Figure 1.1 shows a classical orbit and a quantum eigenfunction on the same billiard.

1.2 Classical Chaos

Chaos in classical systems is characterized by the Lyapunov exponent λ of a system, which describes how quickly nearby trajectories diverge. It is computed as the long time ratio of the divergence of two initially close paths:

$$\lambda = \lim_{t\to\infty} \lim_{|\boldsymbol{\epsilon}|\to 0} \frac{1}{t} \frac{|f(\mathbf{x_0}, t) - f(\mathbf{x_0} + \boldsymbol{\epsilon}, t)|}{|\boldsymbol{\epsilon}|}$$

From this definition it follows that

$$|f(\mathbf{x_0}, t) - f(\mathbf{x_0} + \boldsymbol{\epsilon}, t)| \approx e^{\lambda t}|\boldsymbol{\epsilon}|$$

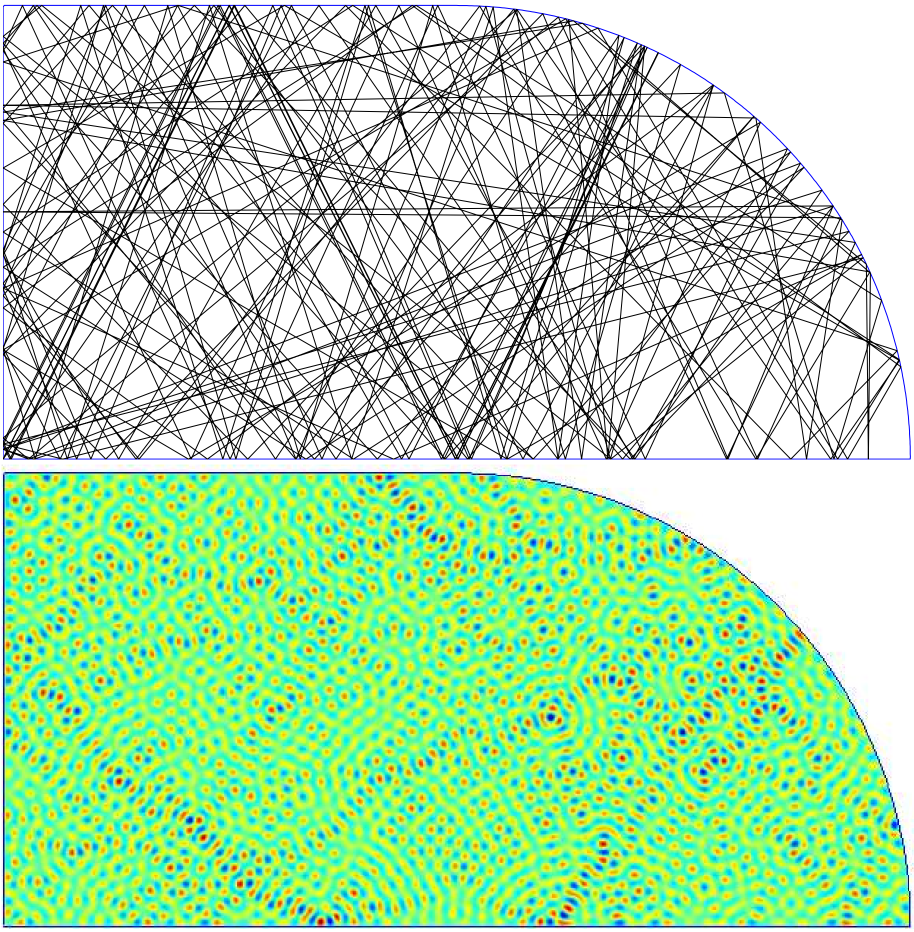

Figure 1.1: Top: A classical orbit in a billiard; bottom: a quantum eigenfunction in the same billiard

for small $\boldsymbol{\epsilon}$ and large t. Thus a tiny change $\boldsymbol{\epsilon}$ in initial conditions produces a change that grows exponentially over time with growth rate λ. Chaotic systems have a positive Lyapunov exponent and this fact makes it difficult to predict long-term behavior because arbitrarily small errors in measurements of initial conditions eventually become large.

1.3 Chaotic Billiards

A compact domain $\Omega \subset \mathbb{R}^2$ is chaotic if a classical point-particle follows a trajectory with a positive Lyapunov exponent. Physically, a domain Ω is represented by a potential function

$$V(\mathbf{r}) = \begin{cases} 0 & \text{if } \mathbf{r} \in \Omega \\ \infty & \text{if } \mathbf{r} \notin \Omega \end{cases}$$

A quantum wave-particle in the presence of such a potential obeys the Schrödinger equation

$$Eu(\mathbf{r}) = -\frac{\hbar^2}{2m}\Delta u(\mathbf{r}) + V(\mathbf{r})u(\mathbf{r})$$

Where $\Delta = \nabla^2 = \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2}$ is the Laplacian differential operator in two dimensions. Setting $\hbar = 2m = 1$ and enforcing continuity of $u(\mathbf{r})$ at the domain boundary $\Gamma = \partial\Omega$ simplifies this to the Helmholtz equation

$$\begin{cases} (\Delta + k^2)u(\mathbf{r}) = 0 & \text{if } \mathbf{r} \in \Omega \\ u(\mathbf{r}) = 0 & \text{if } \mathbf{r} \in \Gamma \end{cases} \tag{1.1}$$

where $k^2 = E$ is the energy of the eigenfunction $u(\mathbf{r})$ and corresponds to the kinetic energy of the quantum wave-particle.

We focus our investigation on to billard shapes: the generalized rectangular Sinai billiard and the Stadium billiard. In both cases we desymmetrize the billard shapes by considering only a quarter of the full shape. This restricts our basis set to functions that are odd as a function of x and y, i.e., $f(-x, y) = f(x, -y) = -f(x, y)$. The desymmetrized billiards are shown in figure 1.2

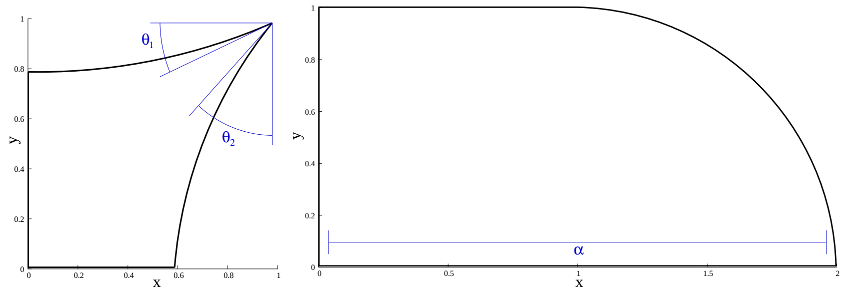

Figure 1.2: Left: quarter generalized rectangular Sinai billiard; right: quarter stadium billiard

The Sinai billiard is constructed from circular arcs that meet at $(1, 1)$ and is parameterized by two angles, θ_1 and θ_2, the angles from horizonal and vertical, respectively, of the arcs at $(1, 1)$. The Sinai billiard is said to demonstrate "hard chaos" becase there are no stable orbits.

The stadium billiard is construced from a rectangular region and a quarter circle and is parameterized by the the horizontal length of the billiard α. Trajectories with vertical momentum in the rectangular region are stable but such orbits occupy a measure zero subset of phase space and the quarter stadium still demonstrates chaotic properties.

Chapter 2

Methods

2.1 Scaling Method

2.1.1 Theory

Vergini and Saraceno [2] developed a method of computing high energy eigenfunctions of chaotic billiards using a scaling method. The method simultaneously finds all eigenfunctions $\phi_i(\mathbf{r})$ with energy in a given window $[E-\Delta E, E+\Delta E]$ for some $E = k^2$ by scaling each eigenfunction. The scaled eigenfunctions $\chi_i(k, \mathbf{r})$ are computed as

$$\chi_i(k, \mathbf{r}) = \phi_i\left(\frac{k}{k_i}\mathbf{r}\right) = \phi_i\left(\mathbf{r} + \frac{\omega_i}{k_i}\mathbf{r}\right)$$

where $\omega_i = k - k_i$. This scaling causes all eigenfunctions to fall approximately in a linear subspace of a single basis set $\{\xi_l\}_{l=1}^{B}$, that is

$$\chi_i(k, \mathbf{r}) = \sum_{l=1}^{B} X_{li}\xi_l(k, \mathbf{r}) + \epsilon_i(\mathbf{r})$$

where $\epsilon_i(\mathbf{r}) \ll 1$ for sufficiently large B, the number of basis functions used. Values of B of approximately $1.5\frac{k|\Gamma|}{\pi}$ have been shown to produce $\epsilon < 10^{-4}$. This single basis set provides a significant efficiency gain over prior methods because many eigenfunctions can be found by solving a single linear system. The basis functions $\xi_l(k, \mathbf{r})$ depend on the billiard shape being used. For the quarter generalized rectangular Sinai billiard the basis set consists of irregular Bessel functions (which satisfy $(\Delta + k^2)\xi_l(k, \mathbf{r}) = 0$) placed along Γ^+, the set of points ouside Ω whose nearest distance to Γ is D where kD is taken to be 7 so that D is apprriximately one wavelength. For the quarter stadium, the basis set is plane waves with orientations evently spaced around the circle. The scaling method runs in $O(B^3) = O(k^3)$ time. [1]

2.1.2 Run Time

The scaling method only computes coefficients of basis functions, which must be evaluated in order to obtain eigenfunction values at arbitrary $\mathbf{r} \in \Omega$. However, because all eigenfunctions in the energy window use the same basis functions, each basis function only needs to be evaluated once at each point, regardless of the number n of eigenfunctions in the energy window. Thus NB basis function evaluations are required. Computing eigenfunctions is then just a matter of multiplying the basis function values at each point by the coefficients for that eigenfunction, which requires a total of nNB multiplications.

In order to maintain accuracy when scaling eigenfunctions we must use relatively small energy windows, typically getting $n \sim 20$ eigenfunctions per call. For both billiards considered here $B \sim k$. Thus, in practice $N \gg n$ and $N \gg B$ so it is primarily N (which is determined by h) that determines the time to compute eigenfunctions. Additionally, we find that the time to evaluate eigenfunctions dominates total computation time [TODO: need timing data].

2.2 Counting Nodal Domains

Eigenfunctions are sampled on a regular grid; each point in this grid will be referred to as a "pixel." Nodal domains are counted by exploring domains pixel-by-pixel, marking each pixel as "counted" once it has been seen. The searching algorithm usded to explore each domain is a hybrid depth- and breadth-first method where for each pixel, the sign of each neighboring pixel is compared to the sign of the nodal domain and if the sign matches, the neighboring pixel is pushed onto a stack. Exploration then continues by popping a pixel off of the stack. This method was chosen so that the signs of all four neighbors (above, below, right, and left) are known when that pixel is being examined, which is necessary for the adaptive interpolation scheme described below.

Letting N be the number of points the eigenfunction is sampled at, this method has computational complexity $O(N)$. This is because the method performs a fixed number of comparisons for each pixel plus $O(N)$ total comparisons searching for an unseen nodal domain after a nodal domain has been explored. This algorithm uses an array of N integers to store, for each pixel, whether it has been counted, which nodal domain it is in, whether or not it is within the boundary of the billiard, and additional information relating to the interpolation method described below. In addition, this method uses a dynamically sized array as a stack whose size is (loosely) bounded above by the number of pixels in the nodal domain being explored.

2.3 Interpolation

Because sampling eigenfunctions is expensive (requiring evaluation of B = O(k) basis functions), we are limited by the total number of pixels N which scales like h^{-2} where h is the distance between adjacent sample points, or the width and height of a pixel. As a consequence, we must work with relatively coarsely sampled eigenfunctions, causing us to encounter scenarios where the connectivity of nodal domains is ambiguous (Fig. 2.1).

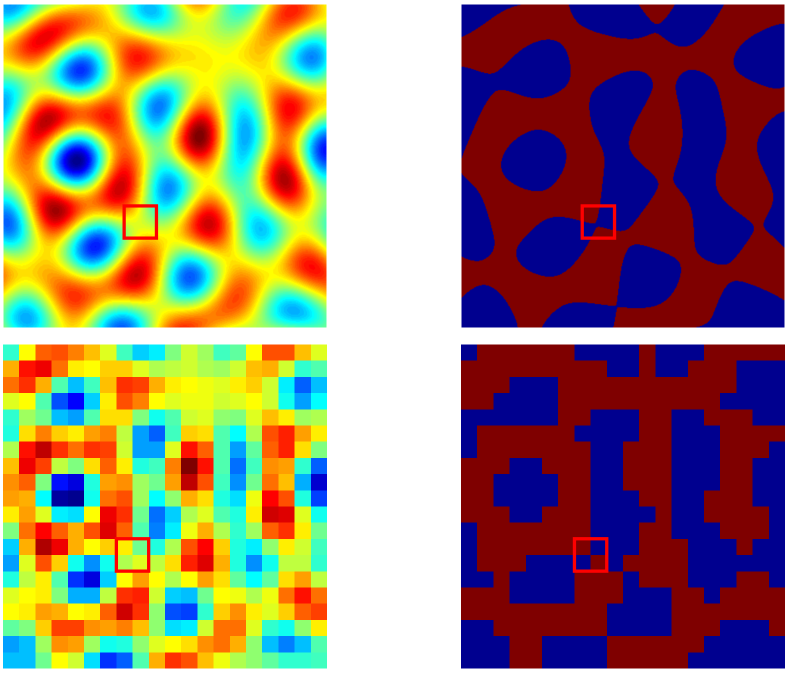

Figure 2.1: An ambiguity in nodal domain connectivity due to coarse sampling. The ambiguous region is shown with a red box. Top left: high resolution image of an eigenfunction; top right: nodal domains of the same eigenfunction; bottom left: low resolution image of the same eigenfunction; bottom right: nodal domains of low resolution eigenfunction

We resolve such ambiguities by performing an interpolation in the ambiguous region. We interpolate with the functions

$$J_n(kr)\sin(n\theta)$$

and

$$J_n(kr)\cos(n\theta)$$

where $J_n(r)$ is a regular Bessel function. These functions form a complete orthonormal basis for solutions of (1.1) (see A). We fix a value M to be the order of the highest order Bessel function, restricting our basis to the finite set

$$\xi_i(r,\theta) = \begin{cases} J_i(kr) & \text{if } i = 0 \\ J_i(kr)\sin(i\theta) & \text{if } 1 \le i \le M \\ J_{i-M}(kr)\cos((i-M)\theta) & \text{if } M+1 \le i \le 2M \end{cases} \qquad (2.1)$$

We construct a surrogate function

$$\tilde{u}(\mathbf{r}) = \sum_{i=0}^{2M} c_i \xi_i(\mathbf{r})$$

as a local approximation to the eigenfunction $u(\mathbf{r})$ by fitting the coeffiecients $c_i \in \mathbb{R}$ to minimize the error $\|u - \tilde{u}\|_2$. This function can then be sampled at a higher resolution within the region in question. We define the sampling ratio of this surrogate function to the original eigenfunction to be ρ.

Interpolation occurs only between the central four pixels but uses surrounding values to fit the coefficients c_i. The selection of which surrounding values to use is termed a stencil. After testing various stencil shapes we selected one using 24 points consisting of a four by four square with two additional points on each side of the square. See B for a discussion of the choice of stencil, M, and h.

2.3.1 Numerical Implementation

We can construct a single matrix to transform values on a stencil to interpolated values in the center of the stencil, performing the process described above with a single matrix multiplication. We construct the interpolation matrix as

$$U = LH^+$$

Where L is a matrix which contains evaluations of Bessel functions at low resolution, H contains evalutions of Bessel functions at high resolution and $^+$ denotes the pseudoinverse. Specifically,

$$L_{ij} = \begin{cases} J_j(r_i) & \text{for } j = 0 \text{ and } 0 \leq i < T \\ J_j(r_i)\sin(j\theta_i) & \text{for } 1 \leq j \leq M \text{ and } 0 \leq i < T \\ J_{j-M}(r_i)\cos((j-M)\theta_i) & \text{for } M+1 \leq j \leq 2M+1 \text{ and } 0 \leq i < T \end{cases}$$

and

$$U_{ij} = \begin{cases} J_j(r_i) & \text{for } j = 0 \text{ and } 0 \leq i < \gamma \\ J_j(r_i)\sin(j\theta_i) & \text{for } 1 \leq j \leq M \text{ and } 0 \leq i < \gamma \\ J_{j-M}(r_i)\cos((j-M)\theta_i) & \text{for } M+1 \leq j \leq 2M+1 \text{ and } 0 \leq i < \gamma \end{cases}$$

where T is the number of points in the stencil, $\gamma = (\rho + 1)^2$ and (r_i, θ_i) are points in the stencil expressed in polar coordinates.

The pseudoinverse H^+ is computed using a singular value decomposition as follows,

$$H^+ = V\Sigma^+U^*$$

where $H = U^*\Sigma V$ is a singular value decompostion of H and

$$\Sigma_{ii}^+ = \begin{cases} \Sigma_{ii}^{-1} & \text{if } \Sigma_{ii} > \epsilon \\ \Sigma_{ii} & \text{otherwise} \end{cases}$$

where $\epsilon = \gamma\epsilon_{double}\Sigma_{11}$ where ϵ_{double} is the difference between 1 and the smallest IEEE double precision floating point number greater than 1. Singular values less than ϵ are considered to be zero within working precision.

2.3.2 Code Implemenation

A region is interpolated if and only if, when counting nodal domains, we encounter a point whose sign matches that of a point diagonally adjacent to it, but differs from the signs of the two points adjacent to both it and its diagonal neighbor (Fig. 2.2). In such a case we fill a vector $\mathbf{v}$ with the eigenfunction values at the stencil points surrounding the four points comprising the ambiguity. We then compute $\mathbf{w} = U\mathbf{v}$ where U is the interpolation matrix described above. This vector $\mathbf{w}$ contains estimated eigenfunction values with a spacing of $\frac{h}{\rho}$ between the four points comprising the ambigious region. We can use these values to determine the connectivity of the nodal domains by traversing pixel-by-pixel from the top-left pixel, in the same manner as above, until we either reach the bottom-right pixel or finish exploring the nodal domain. In the former case the nodal domain containing the top-left pixel connects to the nodal domain containing the bottom-right pixel and in the latter case the nodal domain containing the top-right pixel connects to the nodal domain containing the bottom-left pixel.

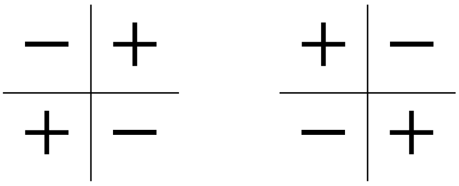

Figure 2.2: Configurations that will result in interpolation

Chapter 3

Results

Appendix A

General Solution of the Helmholtz Equation

Here we show that the functions $J_n(kr)\sin(n\theta)$ and $J_n(kr)\cos(n\theta)$ form a complete orthonormal basis of solutions of (1.1).

In polar coordinates,

$$\Delta = \frac{1}{r}\partial_r(r\partial_r) + \frac{1}{r^2}\partial_{\theta\theta}$$

Thus, (1.1) can be expressed as

$$u_{rr}(r,\theta) + \frac{1}{r}u_r(r,\theta) + \frac{1}{r^2}u_{\theta\theta}(r,\theta) + k^2 u(r,\theta) = 0$$

Using separation of variables we attempt solutions of the form

$$u(r,\theta) = R(r)\Theta(\theta)$$

where $\Theta(\theta)$ is periodic with period 2π. This gives

$$\Theta''(\theta) + n^2\Theta(\theta) = 0$$

and

$$r^2 R''(r) + rR'(r) + r^2k^2R(r) - n^2R(r) = 0$$

The periodicity of $\Theta(\theta)$ requires that

$$\Theta(\theta) = \alpha\sin(n\theta) + \beta\cos(n\theta)$$

where $n \in \mathbb{Z}$. The radial differential equation is known as Bessel's equation and has solutions

$$R(r) = J_n(kr)$$

where $k \in \mathbb{R}$ is allowed to take discrete values determined by boundary conditions.

Thus, the general solution of (1.1) can be expressed as a sum of the form

$$b_0 J_0(kr) + \sum_{n=1}^{\infty} a_n J_n(kr) \sin n\theta + b_n J_n kr \cos n\theta \tag{A.1}$$

Appendix B

Interpolation Parameter Selection

In this section, we discuss the various stencils, M, and α values that were considered and how we picked our particular values. Only stencil shapes that which are symmetric about the four central points were considered. The four shapes considered are shown in figure B.1. The most important consideration in choosing a stencil was the accuracy of the interpolation. The size of the stencil affects the computational cost of interpolation but this difference is trivial.

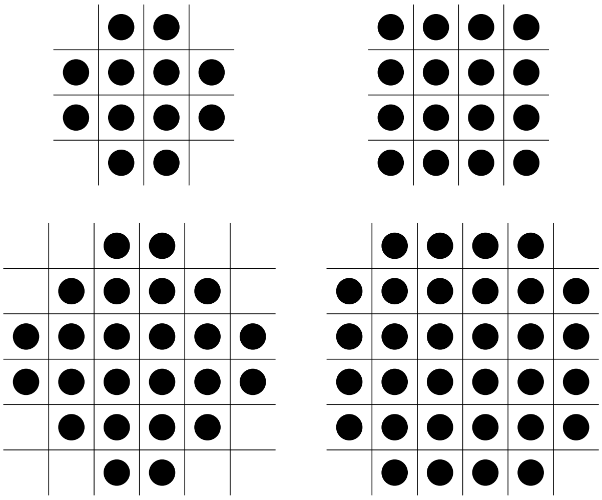

Figure B.1: The four stencil shapes considered for interpolation

The accuracy of interpolation also depends on M and $\alpha = kh$, which must be considered simultaneously with the choice of stencil. Figure B.2 shows a comparison of the infinity norm of the interpolation error over various values of M and α for each of the stencils shown above.

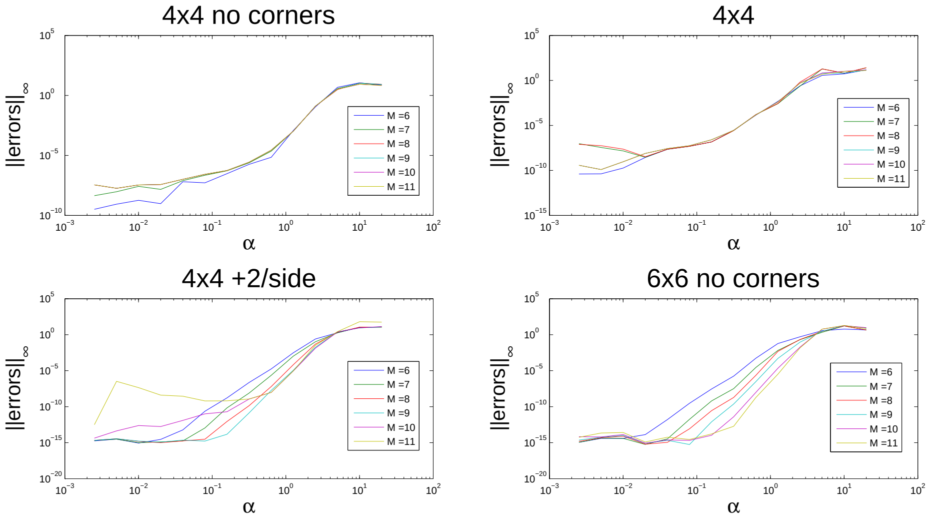

Figure B.2: Comparison of interpolation error norms over M and α for each stencil.

Bibliography

[1] A. Barnett. Asymptotic rate of quantum ergodicity in chaotic euclidean billiards. *Communications in Pure and Applied Mathematics*, 59(10):1457–1488, 2006.

[2] E. Vergini. Calculation by scaling of highly excited states of billiards. *Physical Review E*, (3):2204–2207, 1995.

[3] I. Wigman. On the nodal line of random and deterministic laplace eigenfunctions. 2011.