SJÄLVSTÄNDIGA ARBETEN I MATEMATIK

MATEMATISKA INSTITUTIONEN, STOCKHOLMS UNIVERSITET

Spectra of Quantum Graphs

av

Gabriela Malenová

2012 - No 26

Spectra of Quantum Graphs

Gabriela Malenová

Självständigt arbete i matematik 30 högskolepoäng, Avancerad nivå

Handledare: Pavel Kurasov

2012

DIPLOMA WORK

Spectra Of Quantum Graphs

Gabriela Malenová
Czech Technical University in Prague
Faculty of Nuclear Sciences and Physical Engineering

Supervision:

Pavel Kurasov
Department of Mathematics, Stockholm University,
Stockholm, Sweden

David Krejčiřík
Nuclear Physics Institute in Řež, Academy of Sciences,
Prague, Czech Republic

Abstract

Quantum graph is a network structure determined by:

1. a metric graph consisting of sets of edges and vertices,
2. a differential operator acting on the edges,
3. matching and boundary conditions on internal and external vertices respectively.

Since the spectra of quantum graphs can be calculated analytically in a few special cases only, numerical methods have to be employed. Spectral methods based on Galerkin tau-methods appear to be the most convenient for that purpose. The code in MATLAB environment has been evolved for computing eigenvalues of the graph. Employing numerics, we obtain extensive computational data that may be helpful for understanding fine spectral properties of quantum graphs.

In particular, the spectral gap, i. e. the second eigenvalue of the Laplacian, has been closely investigated. In spite it bears some similar characteristics to discrete graphs we show that unlike in the discrete case, cutting off an edge does not necessarily mean the second eigenvalue increases. Another important result says that a string has always the lowest spectral gap among all graphs of the same total length.

Acknowledgments

Special thanks to:

Pavel Kurasov, for excellent help, advice and mentoring of the project, and for the opportunity to work with him on the research.

David Krejčiřík, for mentoring the thesis, kind support and for carefully reading the drafts.

Erik Wernersson, for valuable help while struggling with the numerics.

Miloš Tater, for careful reading of the thesis.

Contents

1 Introduction

Remarkable progress in the nanotechnology has been made in the last decades. It enabled one to exhibit quantum phenomena in the nanodevices because their typical length is comparable to the atom size. This raised the demand on mathematical studies of quantum networks since they may be used to model such systems.

The origin of quantum graph theory may be traced back to 80's when the initial concept has been introduced (see [6] and references therein). In the recent years, articles related to this topic are published on a regular basis as the concept gained enormous popularity. [5], [12], [11] are counted among the crucial papers. Furthermore, we refer to some significant papers: [14], [13], [17].

More specifically, quantum graphs consist of a *metric graph* Γ, i. e. linear network-shaped structure nesting set of edges $\mathbf{E}$ and vertices $\mathbf{V}$, a *differential operator* acting on the edges with *matching conditions* imposed at the vertices. An intuitive quantum graph model employs the standard Laplacian, i. e. Laplacian on $H^2(\Gamma\backslash\mathbf{V})$ satisfying the *standard matching conditions* at each vertex:

$$\begin{cases} \text{continuity of the functions} \\ \text{the sum of normal derivatives is zero.} \end{cases}$$

This guarantees that the Laplacian is self-adjoint on the graph Γ. More precise definition is provided in Section 2.

In this thesis, we consider compact quantum graphs. In general, it is not possible to analytically find the spectrum since the number of explicitly solvable models is restricted. Some of them, above all the string, star, loop and lasso graphs, are presented in Section 3.

In more complicated cases, the numerical methods have to be applied. In Section 4, the spectral method approach is described which enables us to compute spectrum of the standard Laplacian on arbitrary equilateral quantum graph providing its incidence matrix in the input only. Spectral methods based on Chebyshev polynomials base decomposition grant excellent accuracy.

Once having a tool computing the spectra in hand, we drew our attention to the inverse problems. As a first application, we computed the Euler characteristics derived from the *trace formula* in Subsection 5.1. This gives us the feeling about the number of terms in the sequence that are necessary for achieving requested accuracy.

The second eigenvalue of the standard Laplacian denoted by λ_1, sometimes called the *spectral gap*, was extensively studied for discrete graphs, see [7], [9]. The authors claim λ_1 being the measure of *synchronizability and robustness* of a discrete graph. This evokes one to introduce the same concept for quantum graphs.

Based on observations regarding the second eigenvalue on compact graphs (we know that the first eigenvalue is always 0), some theorems have been stated in Subsection 5.2. We proved that unlike in the discrete case, λ_1 is not granted to be monotonously depending on the number of edges. Namely, we found a

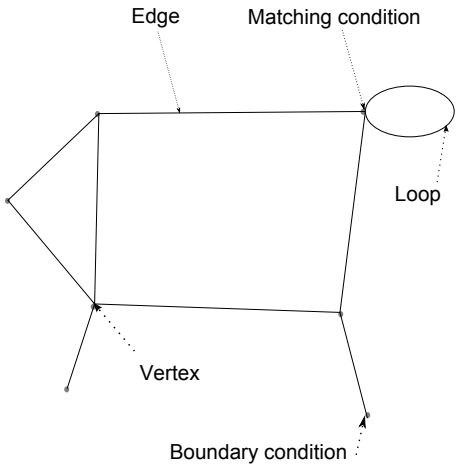

Figure 1: General metric graph.

sufficient, but not necessary condition for the second eigenvalue to grow while cutting off one edge.

Finally, the proof that among the graphs with the same total length the string graph has the lowest spectral gap is provided. We demonstrate this behavior on star graph.

2 Quantum graphs

Rigorous definition of a quantum graph, i. e. Schrödinger operator on the metric graph, contains three main parts:

1. metric graph,
2. differential operator acting on the edges,
3. matching and boundary conditions at internal and external vertices respectively.

These conditions are not completely independent and their connection is explained below. The definitions are taken from the draft of the book by Pavel Kurasov [14].

2.1 Metric graph

In the general sense, a graph is said to be a finite set of edges and vertices (Figure 1). The edges may have finite or infinite length. More precisely, let us define the set $\{E_n\}_{n=1}^N$ of N compact or semi-infinite intervals E_n, each one of them being a subset of $\mathbb{R}$, as:

$$E_n = \begin{cases} [x_{2n-1}, x_{2n}], & n = 1, 2, \ldots, N_c \\ [x_{2n-1}, \infty), & n = N_c + 1, \ldots, N_c + N_i = N, \end{cases}$$

where N_c, respectively N_i, denotes the number of compact, respectively infinite, intervals. The intervals E_n are called edges.

Let us define the set $\mathbf{V}$ of all endpoints

$$\mathbf{V} = \{x_{2n-1}, x_{2n}\}_{n=1}^{N_c} \cup \{x_{2n-1}\}_{n=N_c+1}^{N},$$

and its arbitrary partition into M equivalence classes V_m, $m = 1, 2, \ldots, M$, called vertices. The equivalence classes have the following properties:

$$V = V_1 \cup V_2 \cup \ldots \cup V_m$$
$$V_m \cap V_{m'} = \emptyset, \quad \text{when} \quad m \neq m'.$$

The endpoints belonging to the same equivalence class will be identified

$$x \sim y \Leftrightarrow \begin{bmatrix} \exists n : x, y \in E_n \ \& \ x = y, \\ \exists m : x, y \in V_m. \end{bmatrix}$$

Definition 1. Let us have N edges E_n and a set of M disjoint vertices V_m. Then the corresponding metric graph Γ is the union of all edges with the endpoints belonging to the same vertex identified

$$\Gamma = \bigcup_{n=1}^{N} E_n|_{x \sim y}.$$

The number v_m of elements in the class V_m will be called the *valence* of V_m.

We will mainly concentrate on compact graphs which occur when $N_i = 0$, i. e. all the edges are of finite length and $N = N_c$. Let us consider a complex-valued function u defined on the graph. Then the corresponding Hilbert space yields

$$L_2(\Gamma) = \bigoplus \sum_{j=1}^{N} L_2(E_j).$$

Even if the endpoints coincide, one may define the boundary value of the function as a limit

$$u(x_j) = \lim_{x \to x_j} u(x).$$

The normal derivatives in the endpoints follow the convention that the limits are taken in the direction pointing inside the respective interval, i. e.:

$$\partial_n u(x_j) = \begin{cases} \lim_{x \to x_j} \frac{d}{dx} u(x), & x_j \quad \text{is the left end point}, \\ -\lim_{x \to x_j} \frac{d}{dx} u(x), & x_j \quad \text{is the right end point} \end{cases} . \qquad (1)$$

2.2 Differential operator

To properly implement the dynamics of waves on the graph, one introduces a differential operator. In general, magnetic Schrödinger operator

$$L_{q,a} = \left(i\frac{d}{dx} + a(x) \right)^2 + q(x), \tag{2}$$

is a standard choice for describing quantum phenomena, where a denotes the magnetic potential and q the electric potential respectively. More precisely, we assume $a(x), q(x) \in \mathbb{R}$ satisfying:

1. $q \in L_2(\Gamma)$,
2. $\int_\Gamma (1 + |x|) \cdot |q(x)| dx < \infty$,
3. $a \in C^1(\Gamma)$.

Putting the magnetic potential equal to zero $a = 0$ we obtain Schrödinger operator

$$L_q = -\frac{d^2}{dx^2} + q(x). \tag{3}$$

Setting the potentials $a = 0 = q$ we get the Laplace operator describing the free motion:

$$L = -\frac{d^2}{dx^2}. \tag{4}$$

Hereby we list some types of domains the Laplacian may be defined on. Firstly, let us consider the *maximal operator* $L^{\max}$ corresponding to (4) defined on the domain $D(L^{\max}) = H^2(\Gamma \backslash \mathbf{V})$, where H^2 denotes the Sobolev space of all square integrable functions having square integrable first and second derivatives. This domain may be written in the decomposed fashion as the sum of Sobolev spaces on the intervals E_n

$$D(L^{\max}) = \bigoplus \sum_{n=1}^{N} H^2(E_n),$$

independently on how the edges are connected to each other. Similarly, the operator $L^{\max}$ can be decomposed as

$$L^{\max} = \bigoplus \sum_{n=1}^{N} L^n,$$

where L^n is given by (4) on the domain $H^2(E_n)$.

Similar relations hold for the *minimal operator* $L^{\min}$ defined on $C_0^\infty(\Gamma \backslash \mathbf{V})$.

2.3 Matching conditions

The vertices may be divided into two groups- the *internal* vertices which have valence greater than one, in other words there are at least two edges meeting in the vertex. The subset of all conditions introduced at the internal points is called *matching conditions.* The other group is made up of vertices of valence equal to one called *boundary* vertices and *boundary conditions* are enforced there (see Figure 1).

The maximal operator $L^{\max}$ is neither self-adjoint nor symmetric. The self-adjointness may be achieved by imposing certain conditions on $u, v \in D(L^{\max})$:

$$\begin{aligned}\langle L^{\max}u, v\rangle - \langle u, L^{\max}v, \rangle &= \sum_{n=1}^{N}\left(\int_{E_n} -u''(x)\overline{v(x)}\,dx + \int_{E_n} u(x)\overline{v''(x)}\,dx\right) = \\ &= \sum_{x_j\in\mathbf{V}}\left(\partial_n u(x_j)\overline{v(x_j)} - u(x_j)\overline{\partial_n v(x_j)}\right), \qquad (5)\end{aligned}$$

where the normal derivative at the endpoints is recalled from (1). Thus the operator $L^{\max}$ to be symmetric requires the boundary form in (5) being equal to 0. Then the operator possesses real spectrum.

What comes to one's mind at first is to require that the functions are equal to zero at the vertices.

Definition 2. The *Dirichlet Laplace operator* L^D is defined by the differential expression (4) on the Sobolev space $H^2(\Gamma\backslash\mathbf{V}) \hookrightarrow C^1(\Gamma\backslash V)$ satisfying the Dirichlet conditions

$$u(x_j) = 0,\ x_j \in \mathbf{V},$$

at all end points.

The Dirichlet Laplacian, may by presented in the decomposed way as:

$$L^D = \bigoplus \sum_{n=1}^{N} L^{n,D},$$

where $L^{n,D}$ is the differential operator (4) restricted to the set of all functions from the Sobolev space $H^2(E_n)$ satisfying the Dirichlet boundary conditions at endpoints. However, this is not an interesting case, since such an operator builds a graph model where all the edges are separated from each other and behave independently.

Another way how to impose conditions (5) without separating the edges is through *standard matching and boundary conditions*[1] introduced in each vertex V_m:

$$\begin{cases} u \text{ is continuous at } V_m \\ \sum_{x_j\in V_m} \partial_n u(x_j) = 0. \end{cases} \qquad (6)$$

[1]Standard matching conditions are sometimes called *Free, Neumann or Kirchhoff conditions.*

For boundary vertices this simply yields the Neumann boundary condition

$$\partial_n u(x_j) = 0,\ x_j \in V_m,\ V_m \text{ is a boundary vertex.}$$

If there were two edges connected in a vertex this would imply nothing else than the continuity of the function and its first derivative. In that case, the vertex may be removed and the two intervals may be substituted by one of the sum of original sizes. Matching conditions give us a tool for setting up a self-adjoint operator called *standard Laplace operator* L^{st}:

Definition 3. The standard Laplace operator L^{st} is defined by the differential expression (4) on the domain $H^2(\Gamma\backslash\mathbf{V})$ satisfying the standard matching conditions (6) at all vertices.

2.4 Elementary spectral properties

Since we consider compact graphs formed by finitely many edges, spectral properties may be characterized by the following theorem.

Theorem 2.1. *Let Γ be a finite compact graph and $L_q = L^{\mathrm{st}} + q$ the corresponding Schrödinger operator* (3)*. Then the spectrum is purely discrete and consists of infinite sequence of real eigenvalues with one accumulation point $+\infty$. The proof may be found in [14].*

Note, that the standard Laplacian as well as the Dirichlet Laplacian are in fact the extensions of the symmetric operator defined by the same formula (4) on the domain of all continuous functions from $H^2(\Gamma\backslash\mathbf{V})$ subject to the following conditions:

$$\begin{cases} u(V_m) = 0 \\ \sum_{x_j \in V_m} \partial u(x_j) = 0 \end{cases},$$

for all vertices.

Some important facts regarding standard Laplacian remain to be proven. Above all, we always precisely know how the first eigenvalue looks like.

Proposition 2.2. *$\lambda_0 = 0$ is the first eigenvalue of the standard Laplace operator L^{st} on finite compact graph Γ with multiplicity n equal to number of connected components $\Gamma = \Gamma_1 \cup \ldots \cup \Gamma_n$. The i-th eigenvector is equal to $1 \in L_2(\Gamma_i)$ and zero elsewhere.*

Proof. The eigenvectors corresponding to $\lambda_0 = 0$ are linear functions since they satisfy

$$-\psi'' = 0, \qquad \psi(x) = \alpha_n x + \beta_n,$$

on each edge E_n. Application of the standard matching conditions (6) preserves continuity of the eigenvectors which makes their maxima attainable on the endpoints only. Say, we achieved the global maximum. Sum of the derivatives is zero at each node, but in this point, they have to be negative (due to the maximum). This necessarily implies, they are identically equal to zero. All in

all, ψ is constant on every edge attached to the maximum. We conclude that the function is constant on all such edges. Consider another neighboring vertex and repeat the arguments. We continue in this way until the whole connected component is covered. This brings us to the claim, that the spectral multiplicity of the eigenvalue $\lambda_0 = 0$ is the number of connected components. □

However, it is not always necessary to compute the whole infinite sequence of eigenvalues. For example if the lengths of all edges are integer multiples of one basic length then the spectrum is periodic and it is enough to calculate just the first few eigenvalues to know the whole spectrum.

Proposition 2.3. *Provided $k^2 \neq 0$ is an eigenvalue of the Schrödinger operator L^{st} (4) satisfying the quadratic form (5) is equal to zero on a graph Γ consisting of basic lengths ($l_j = m_j\Delta$), then $(k + \frac{2\pi}{\Delta})^2$ also belongs to the spectrum.*

Proof. Let us consider $k^2 \in \sigma(L^{\mathrm{st}}(\Gamma))$. Then the eigenvector ψ restricted to the n-th edge $[x_{2n-1}, x_{2n}]$ is given by (for the derivation see [14]):

$$\psi(x)|_{E_n} = a_{2n-1}e^{ik|x-x_{2n-1}|} + a_{2n}e^{ik|x-x_{2n}|}.$$

Shifting the frequency

$$k \longrightarrow k + \frac{2\pi}{\Delta},$$

we obtain new function $\widetilde{\psi}$:

$$\widetilde{\psi}(x)|_{E_n} = a_{2n-1}e^{i(k+\frac{2\pi}{\Delta})|x-x_{2n-1}|} + a_{2n}e^{i(k+\frac{2\pi}{\Delta})|x-x_{2n}|}.$$

Comparing the boundary values we get

$$\widetilde{\psi}(x_{2n}) = a_{2n-1}e^{ikm_n\Delta} + a_{2n} = \psi(x_{2n}).$$

Similarly,

$$\widetilde{\psi}(x_{2n-1}) = \psi(x_{2n-1}).$$

Analogously, the derivatives are carried out:

$$\widetilde{\psi}'(x_{2n-1}) = i\left(k + \frac{2\pi}{\Delta}\right)(a_{2n-1} + a_{2n}e^{ikm_n\Delta}) = \left(1 + \frac{2\pi}{\Delta k}\right)\psi'(x_{2n-1}),$$

and

$$\widetilde{\psi}'(x_{2n}) = \left(1 + \frac{2\pi}{\Delta k}\right)\psi'(x_{2n}).$$

This means, $\widetilde{\psi}$ is an eigenfunction corresponding to the eigenvalue $\left(k + \frac{2\pi}{\Delta}\right)^2$. □

3 Explicit solutions

Let us first introduce some elementary cases of quantum graphs where the spectrum can be calculated explicitly. The most natural starting point is to consider the Laplacian on a single interval.

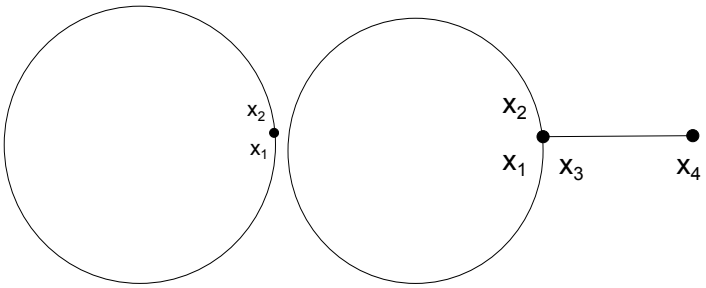

Figure 2: Loop and lasso graphs.

3.1 Interval

We have the Dirichlet Laplacian (Def 2) on a single interval $[x_1, x_2]$. Consequently, we may reparametrize it to $[0, l]$. Thus we are to solve the problem

$$\begin{cases} -u'' = \lambda u \\ u(0) = 0,\ u(l) = 0. \end{cases}$$

Any solution to the differential equation can be obtained in the form

$$u(x) = A\cos kx + B\sin kx, \quad A, B \in \mathbb{C}, \tag{7}$$

where $k^2 = \lambda$ which implies the eigenvalues being in the form of infinite sequence

$$\lambda_n = \frac{\pi^2}{l^2}n^2, \quad n = 1, 2, \ldots,$$

with the eigenvectors

$$u(x) = \sin\frac{\pi n x}{l}, \quad n = 1, 2, \ldots.$$

Similar calculations can be carried out for the standard operator (Def 3) on the same interval. Thus we need to solve the problem given by

$$\begin{cases} -u'' = k^2 u \\ u'(0) = 0,\ u'(l) = 0, \end{cases} \tag{8}$$

where the solution form is recalled from (7). Then the constraint to be solved is

$$k\sin kl = 0,$$

whose solution is the sequence

$$\lambda_n = \frac{\pi^2}{l^2}n^2, \quad n = 0, 1, 2, \ldots \tag{9}$$

as before, with the eigenvectors

$$u(x) = \cos\frac{\pi n x}{l}, \quad n = 0, 1, 2, \ldots.$$

Notice, that in addition there is the zero eigenvalue $\lambda_0 = 0$ included as well, with the eigenvector $u(x) = 1$.

3.2 Loop graph

Another case of graph formed by just one edge is a *loop* (Figure 2a). Here, the endpoints are identified and the edge consist of one interval $[x_1, x_2] = [0, l]$. The stationary Schrödinger equation yields

$$-u'' = k^2 u,$$

where $\lambda = k^2$ is an eigenvalue of the differential operator L^{st}. The matching conditions imply that

$$\begin{cases} u(0) = u(l) \\ u'(0) = u'(l) \end{cases}.$$

Plugging it into the Ansatz (7) we arrive at the constraint

$$2k(1 - \cos kl) = 0.$$

The eigenvalue $\lambda = 0$ is a simple eigenvalue with the eigenfunction $u = 1$. The eigenvalues

$$\lambda_n = \frac{4\pi^2}{l^2} n^2, \quad n = 1, 2, \ldots, \tag{10}$$

are of the multiplicity 2. The eigenvectors may be split into two groups according to the criterion whether they are or are not invariant under the change of variables $x \mapsto l - x$. All the even, respectively odd functions are denoted by u_e, respectively u_o, more precisely:

$$u_e(x) = \cos \frac{2\pi}{l} nx, \quad u_o(x) = \sin \frac{2\pi}{l} nx.$$

3.3 Lasso graph

The obvious way to proceed further is to start connecting the intervals together. The *lasso graph* (Figure 2b) is build up by joining one loop with an interval attached. Mathematically, this graph Γ may be defined as a union of two intervals $E_1 = [x_1, x_2]$ and $E_2 = [x_3, x_4]$ with the endpoints x_1, x_2, x_3 identified in one vertex $V_1 = \{x_1, x_2, x_3\}$. In the view of symmetry, it is more convenient to choose the parametrization of edges as follows:

$$[x_1, x_2] = [-l/2, l/2], \quad [x_3, x_4] = [0, L].$$

The operator is invariant under the change of variables

$$J : x \mapsto \begin{cases} -x, & x \in E_1, \\ x, & x \in E_2. \end{cases}$$

This transformation can be lifted to act on functions

$$(Jf)(x) = f(Jx).$$

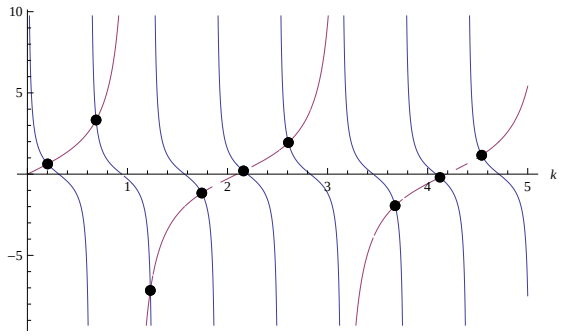

Figure 3: Graphical solution of equation (13) for $L = 5$ and $l = 3$.

We see that the Laplacian is commuting with J

$$JL = LJ,$$

hence the corresponding Laplacian eigenfunctions may be chosen symmetric and antisymmetric with respect to J:

$$Jf_{\text{sym}} = f_{\text{sym}}, \quad Jf_{\text{asym}} = -f_{\text{asym}}.$$

The Laplacian is self-adjoint when defined on functions satisfying the following conditions

$$\begin{cases} u(x_4) = 0, \\ u(x_1) = u(x_2) = u(x_3) \\ u'(x_1) - u'(x_2) + u'(x_3) = 0. \end{cases} \tag{11}$$

Let us first start with the antisymmetric functions. They are necessarily equal to zero on the second interval. On the loop, the eigenfunctions are of the form $u(x) = A \sin kx$ due to the antisymmetry. This requires zero value in the middle point, i. e. the condition

$$A \sin kl/2 = 0, \tag{12}$$

which is satisfied if

$$\lambda_n = \frac{4\pi^2 n^2}{l^2}, \quad n = 1, 2, \ldots.$$

The third condition in (11) obviously holds due to antisymmetry. The symmetric eigenfunctions are of the form

$$u = \begin{cases} D \cos kx, & x \in E_1 \\ C \sin k(x - L), & x \in E_2. \end{cases}$$

To satisfy the second condition in (11), the following constraint has to hold true:

$$D \cos kl/2 = C \sin(-kL).$$

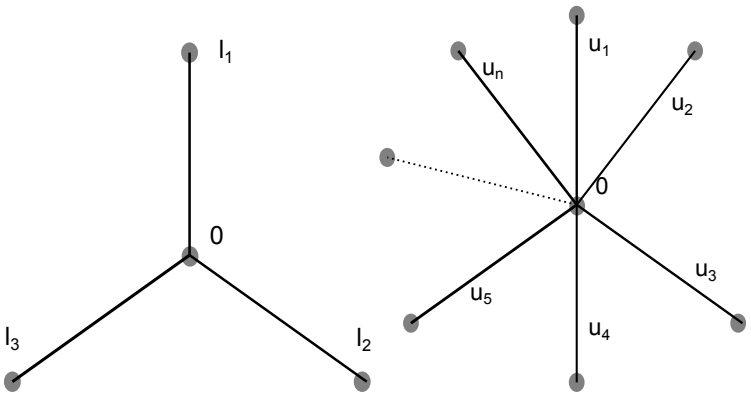

Figure 4: 3-star and n-star graph with parametrization provided zero is in the middle point.

The third condition in (11) implies the equation:

$$2D \sin kl/2 + C \cos kL = 0.$$

The two equations form a 2×2 linear system which has a nontrivial solution if and only if the corresponding determinant is equal to zero:

$$\cot kL = 2 \tan kl/2. \tag{13}$$

The graphical solution for cases $L = 5$ and $l = 3$ is depicted in Figure 3. All in all, joining symmetric with antisymmetric constraint (12) one obtains the eigenvalues condition:

$$(\cot kL - 2 \tan kl/2) \sin kl/2 = 0.$$

Thus the solution can be computed explicitly only in the case L and l are rationally dependent.

3.4 3-star graph

Let us consider a star-shaped graph, namely a set of edges of arbitrary lengths meeting in one central point (see Figure 4, left). The three edges' lengths are denoted by l_1, l_2 and l_3 respectively. The most convenient way of parametrization is to design each edge as $E_n = [0, l_n]$. Thus the solution (applying standard matching conditions) satisfies the following system of equations:

$$\begin{cases} u_1(0) = u_2(0) = u_3(0), \\ u_1'(0) + u_2'(0) + u_3'(0) = 0, \\ u_1'(l_1) = u_2'(l_2) = u_3'(l_3) = 0, \end{cases}$$

where u_j denotes the values of the function u on one of the three intervals. The functions u_j are of the form (7):

$$u_j(x) = A_j \cos k(x - l_j) + B_j \sin k(x - l_j).$$

Applying the conditions mentioned just above, we end up with the matrix equation

$$\underbrace{\begin{pmatrix} \cos kl_1 & -\cos kl_2 & 0 \\ 0 & \cos kl_2 & -\cos kl_3 \\ \sin kl_1 & \sin kl_2 & \sin kl_3 \end{pmatrix}}_{=:M} \begin{pmatrix} A_1 \\ A_2 \\ A_3 \end{pmatrix} = \hat{0}$$

The requirement of the solution to be non-trivial leads to the condition that the determinant is zero:

$$\det M = 0.$$

The equation may be re-written as

$$\begin{aligned} 0 &= \cos kl_1 \cos kl_2 \sin kl_3 + \cos kl_1 \sin kl_2 \cos kl_3 + \sin kl_1 \cos kl_2 \cos kl_3 = \\ &= \cos kl_2 \sin k(l_1 + l_3) + \frac{1}{2} \sin kl_2 [\cos k(l_1 - l_3) + \cos k(l_1 + l_3)], \end{aligned} \tag{14}$$

or similarly, after some algebra:

$$0 = 3 \sin kL + \sin k(-l_1 + l_2 + l_3) + \sin k(l_1 - l_2 + l_3) + \sin k(l_1 + l_2 - l_3),$$

where $L = l_1 + l_2 + l_3$. Solution to this equation may be computed numerically.

Special case of two edges of the star graph having the same length was considered. Here on we set $l_1 = l_3 = l$, which brings us to the constraint

$$\begin{aligned} 0 &= \cos kl(2 \cos kl_2 \sin kl + \sin kl_2 \cos kl) = \\ &= \cos kl(\cos kl_2 \sin kl + \sin k(l + l_2)). \end{aligned}$$

This implies two types of solution. First,

$$\cos kl = 0 \qquad \Longrightarrow \qquad k_n = \frac{\pi}{l} \left(\frac{1}{2} + n \right), \; n = 0, 1, \ldots . \tag{15}$$

The other part has to be evaluated numerically, k is the solution of the following equation:

$$0 = 2 \cos kl_2 \sin kl + \sin kl_2 \cos kl. \tag{16}$$

The solutions from both the above conditions coincide only if there are some $n, m \in \mathbb{Z}$ such that

$$\frac{l_2 - l}{2} = ml - nl_2.$$

3.5 Equilateral star graph

In general, a star graph may be build up from higher number of branches. The computations, as the number rises, are getting excessively large. The only case we are able to solve the problem exactly occurs when all the edges have the same length l.

Let us start with an n-star graph presented in Figure 4. Taking standard matching conditions into account, we require

$$\begin{cases} u_1(0) = u_2(0) = \ldots = u_n(0), \\ \sum_i u_i'(0) = 0, \\ u_1'(l) = u_2'(l) = \ldots = u_n'(l) = 0. \end{cases} \tag{17}$$

We take advantage of the graph being rotationally symmetric with respect to the central node. Let us define the operator of rotation R as

$$R(u_1, u_2, u_3, \ldots, u_n) := (u_2, u_3, \ldots, u_n, u_1).$$

The operators L^{st} and R commute

$$RL^{st} = L^{st}R.$$

This leads to the eigenvalue problem

$$L^{st}(Ru) = \lambda\mu u, \qquad \text{where } Ru = \mu u,$$

with u being the eigenvector of both L^{st} and R since they are self-adjoint.

Due to the fact

$$R^n = 1$$

eigenvalues of R are n-th roots of 1. As we already mentioned, the threshold eigenvalue λ_0 of a standard Laplacian is always 0 and the corresponding eigenvector is $1 \in L_2(\Gamma)$. This is the case of μ_0 equal to one:

$$R(1, 1, 1, \ldots, 1) = 1 \cdot (1, 1, 1, \ldots, 1).$$

The eigenvector corresponding to $\mu_1 = e^{i\frac{2\pi}{n}}$ obeys

$$\begin{aligned} R\left(1, e^{i\frac{2\pi}{n}}, e^{2i\frac{2\pi}{n}}, \ldots, e^{(n-1)i\frac{2\pi}{n}}\right) &= \left(e^{i\frac{2\pi}{n}}, e^{2i\frac{2\pi}{n}}, \ldots, e^{(n-1)i\frac{2\pi}{n}}, 1\right) = \\ &= e^{i\frac{2\pi}{n}}\left(1, e^{i\frac{2\pi}{n}}, e^{2i\frac{2\pi}{n}}, \ldots, e^{(n-1)i\frac{2\pi}{n}}\right). \end{aligned}$$

Similarly, we may proceed further by analogously applying multiple rotations on the vector, henceforth we arrive at

$$\mu_k = e^{ki\frac{2\pi}{n}}$$

and the respective eigenspaces read as follows:

$$\begin{aligned} (1, z, z^2, \ldots z^{n-1}) \times L_2([0, l]), \\ (1, z^2, z^4, \ldots z^{2(n-1)}) \times L_2([0, l]), \\ \vdots \end{aligned}$$

providing $z = e^{i\frac{2\pi}{n}}$. It means that one can look for eigenfunctions of L of the form

$$Rf = z^k f, \quad k = 0, 1, \dots n-1.$$

Setting $k = 0$ gives us symmetric functions while $k = 1, 2, \dots n-1$ defines quasi invariant functions. Functions from the latter class satisfy standard matching conditions only if they are zero at the central vertex. Indeed, from the continuity condition in (17), it follows that

$$u_1(0) = \underbrace{e^{i\frac{2\pi}{n}} u_1(0)}_{u_2(0)} \qquad \Longrightarrow \quad u_1(0) = 0,$$

since $e^{i\frac{2\pi}{n}} \neq 1$.

The derivatives continuity condition in (17) is satisfied due to quasi invariance, indeed:

$$u_1'(0) + u_2'(0) + \dots + u_n'(0) = \left(1 + e^{i\frac{2\pi}{n}} + e^{2i\frac{2\pi}{n}} + \dots + e^{(n-1)i\frac{2\pi}{n}}\right) u_1'(0) =$$
$$= \left(\frac{1 - e^{ni\frac{2\pi}{n}}}{1 - e^{i\frac{2\pi}{n}}}\right) u_1'(0) = 0,$$

and the same holds for its powers. Hence, we have

$$\begin{cases} u_1(0) = 0, \\ u_1'(l) = 0. \end{cases}$$

The solution employs sinus function,

$$u_1(x) = B \sin kx,$$

which after some algebra implies

$$k_n = \frac{\pi}{2l} + \frac{n\pi}{l},$$

whose multiplicity is $n - 1$.

Let us proceed further with the symmetric part. For the derivatives-continuity condition in (17) to be satisfied we need

$$0 = \sum_i u_i'(0) = n u_1'(0),$$

which implies the solution in the form:

$$\begin{cases} u_1'(0) = 0, \\ u_1'(l) = 0, \end{cases}$$

that is resolved by cosine function:

$$u_1(x) = A \cos kx.$$

Hence, the solution yields

$$k_n = \frac{\pi n}{l}.$$

Multiplicity of such an eigenvalue is 1.

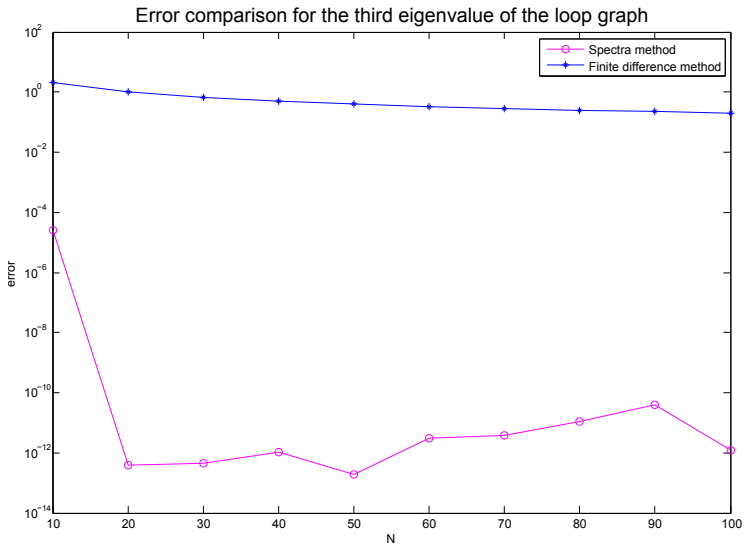

Figure 5: Compare the accuracy rate for spectral method and finite difference method of the first order of the eigenvalue λ_3 in the loop case (where the value is exactly known).

4 Numerical analysis

In the preceding chapter, we have presented some of the explicitly solvable (or nearly explicitly solvable, by transforming into root-finding task) eigenvalue problems. However, their number is very limited. In further proceeding, numerical computation plays an important role. The question arises, which numerical method to choose to compute spectrum of a general equilateral quantum graph. Note, that for simplicity reasons we consider all edges identified with the interval $[-1, 1]$. We will be working in the MATLAB environment.

4.1 Spectral methods

The first method that immediately comes to one's mind is some kind of finite difference formula. MATLAB takes use of sparse matrices, so the codes run in the fraction of seconds. However, the speed of such computation is at the expense of accuracy. From this point of view, the spectral methods are more suitable for problems requiring high order of precision. Spectral accuracy is remarkable, however, there is a price to be paid: full matrices replace sparse matrices, stability restrictions may become more severe, and computer implementations may not be so straightforward.

As the number of grid points N increases, the error for finite difference and finite element scheme typically decreases like $O(N^{-m})$ for some constant m depending on the order of approximation and the smoothness of the solu-

tion. For the spectral method, convergence of the rate $O(N^{-m})$ for *arbitrary* m is achieved, provided the solution is infinitely differentiable, and even faster convergence at a rate $O(c^N)$, $0 < c < 1$ is achieved if the solution is analytic [18].

This behavior is illustrated by Figure 5. The error of spectral and finite difference methods is plotted in the case of loop graph, where the solution (10) is explicitly known. Obviously, finite difference method result is improved very slowly compared to spectral method. Reaching $N = 20$ points, spectral methods achieve the accuracy 10^{-12} where the truncation error does not allow the error to drop more. This is typical spectral accuracy behavior.

4.2 Chebyshev nodes

The eigenvectors of the Schrödinger operator (2) are smooth continuous functions, thus according to [18] it is customary to interpolate them by algebraic polynomials $p(x) = a_0 + a_1 x + \dots a_n x^N$. To avoid Runge phenomenon (oscillating near the endpoints) one introduces the discretization on unevenly spaced points.

Various different sets of points are effective but they shall be distributed asymptotically as $N \to \infty$ with the density per unit length as

$$\text{density } \sim \frac{N}{\pi\sqrt{1-x^2}},$$

which means that they cluster near the endpoints. The canonical interval is $[-1, 1]$. If the differential equation is posed on $[a, b]$, it should first be converted to $[-1, 1]$ through the change of variables

$$x \mapsto \frac{(b-a)x + (b+a)}{2}.$$

One of the node set satisfying the density property on the bounded interval are *Chebyshev points.* There exist more types, the most commonly used ones are so called *Chebyshev Gauss-Lobatto quadrature points* [2]:

$$x_j = \cos\frac{j\pi}{N}, \quad j = 0, 1, \dots, N. \tag{18}$$

In the literature, the *Chebyshev points of the second kind* are sometimes utilized:

$$x_j = \cos\frac{(j-1)\pi}{N-1}, \quad j = 1, 2, \dots, N. \tag{19}$$

Note, that the ordering is defined from right to left.

4.3 Chebyshev polynomials

The Chebyshev points (18) are the roots of so called *Chebyshev polynomials of the first kind*, $T_k(x)$ where $k = 0, 1, \dots$, see Figure 6. Similarly, Chebyshev

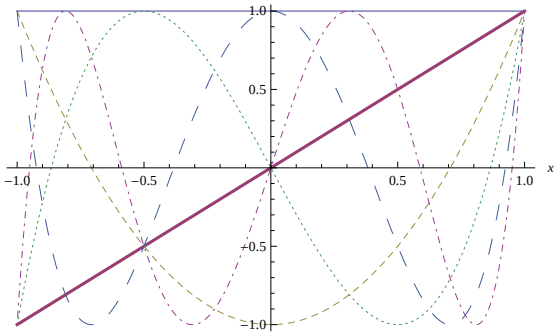

Figure 6: First six Chebyshev polynomials.

nodes (19) are the roots of $T_{k-1}(x)$ on $[-1,1]$. In fact, Chebyshev polynomials are the eigenfunctions of the Sturm-Liouville problem

$$\left(\sqrt{1-x^2}\,T_k'(x)\right)' + \frac{k^2}{\sqrt{1-x^2}}\,T_k(x) = 0.$$

The polynomials may be also given by the recursion relation

$$T_{k+1}(x) = 2xT_k(x) - T_{k-1}(x), \quad T_0(x) = 1, \quad T_1(x) = x.$$

For more details see [2].

Chebyshev polynomials are real and orthogonal with respect to the weight $w(x) = \frac{1}{\sqrt{1-x^2}}$ on $(-1,1)$

$$\int_{-1}^{1} T_n(x)\,T_m(x)\,\frac{dx}{\sqrt{1-x^2}} = \begin{cases} 0, & n \neq m, \\ \pi, & n = m = 0, \\ \pi/2, & n = m \neq 0, \end{cases}$$

and build the basis in the weighted space $L^2_w(-1,1)$. Chebyshev expansion of a function $u \in L^2_w(-1,1)$ is

$$u = \sum_{k=0}^{\infty} \hat{u}_k T_k(x), \quad \hat{u}_k = \frac{2}{\pi c_k} \int_{-1}^{1} u(x) T_k(x) w(x)\,dx,$$

where

$$c_k = \begin{cases} 2, & k = 0, \\ 1, & k \geq 1. \end{cases}$$

4.4 Differentiation matrix

There are two options for to accomplish the differentiation of a function depending on its representation, we can either stay in the transform space $L^2_w(-1,1)$ or express the function in physical space $L^2(-1,1)$. The other way is preferred.

To carry out the computation in the physical space one needs to define its base first. *Characteristic Lagrange polynomials* ψ_l are natural choice- they are unique polynomials that satisfy

$$\psi_l(x_j) = \delta_{jl}, \quad j = 0, \ldots, N.$$

The general expression for such polynomials is

$$\psi_l(x) = \prod_{j \neq l, 0 \leq j, l \leq N} \frac{x - x_j}{x_l - x_j}. \tag{20}$$

For numerical stability reasons, often the Lagrangian polynomials are reformulated in *barycentric form* as

$$\psi_l(x) = \frac{\frac{\lambda_l}{x - x_l}}{\sum_{k=0}^{N} \frac{\lambda_k}{x - x_k}}, \qquad \lambda_l = \frac{1}{\prod_{k \neq l}(x_l - x_k)}. \tag{21}$$

Differentiation in physical space is accomplished by replacing truncation by interpolation. Given a set of $N + 1$ nodes in $[-1, 1]$ the polynomial

$$D_N u = \left(\sum_{l=0}^{N} u(x_l) \psi_l \right)'$$

is called the *Jacobi interpolation derivative* of u. The coefficients are given by $(D_N)_{jl} = \psi_l'(x_j)$, they form the entries of the *first-derivative interpolation matrix* D_N.

In our case it may be shown, that the characteristic Lagrange polynomials (20) at the Chebyshev Gauss-Lobatto points (18) may be expressed as

$$\psi_l(x) = \frac{(-1)^{l+1}(1 - x^2)\, T_N'(x)}{\bar{c}_l N^2 (x - x_l)},$$

where

$$\bar{c}_j = \begin{cases} 2, & j = 0, N, \\ 1, & j = 1, \ldots, N - 1. \end{cases}$$

From this, one gets the derivative interpolation matrix:

$$(D_N)_{jl} = \begin{cases} \frac{\bar{c}_j (-1)^{j+l}}{\bar{c}_l (x_j - x_l)}, & j \neq l, \\ -\frac{x_l}{2(1 - x_l^2)}, & 1 \leq j = l \leq N - 1, \\ \frac{2N^2 + 1}{6}, & j = l = 0, \\ -\frac{2N^2 + 1}{6}, & j = l = N. \end{cases}$$

Numerically more stable code takes advantage of the barycentric formula (21):

$$(D_N)_{jl} = \begin{cases} \frac{\delta_j}{\delta_l} \frac{(-1)^{j+l}}{x_j - x_l}, & j \neq l, \\ -\sum_{i=0, i \neq j}^{N} \frac{\delta_i}{\delta_j} \frac{(-1)^{i+j}}{x_j - x_i}, & j = l, \end{cases} \tag{22}$$

where $\delta_l = 1/2$ if $l = 0$ or N, $\delta_l = 1$ otherwise.

The ready-made function `chebdif.m` by Weideman and Reddy implements the expression similar to (22) [2] on Chebyshev nodes of the second kind (19). The documentation for the MATLAB suite may be found in [19]. The program makes use of the fact, that for spectral methods it holds

$$D_N^{(l)} = (D_N^{(1)})^l, \quad l = 1, 2, \ldots, \tag{23}$$

thus any higher order differentiation matrix can be computed from (22). Moreover, the following facts are incorporated:

1. Making use of the identity $\cos\theta = \sin(\pi/2 - \theta)$ the Chebyshev nodes (19) gain the following form

$$x_k = \sin\left(\frac{\pi(N+1-2k)}{2(N-1)}\right), \quad k = 1, \ldots N,$$

   whose advantage is that it yields nodes which are symmetric about the origin, that is not the case of (19).

2. The differences $x_k - x_l$ may cause the floating point cancelation errors for large N. These differences may be avoided by the use of the trigonometric identity

$$\cos\left(\frac{(k-1)\pi}{N-1}\right) - \cos\left(\frac{(l-1)\pi}{N-1}\right) = 2\sin\left(\frac{\pi(k+l)}{2(N-1)}\right)\sin\left(\frac{\pi(l-k)}{2(N-1)}\right).$$

4.5 Eigenvalue problem

Let us go back to the eigenvalue problem. In the discretized way, we are to solve:

$$-D_N^{(2)}u = \lambda u, \quad u = [u_0, \ldots u_N]^T, \tag{24}$$

where $u_i := u(x_i)$ and $D_N^{(2)}$ is the second order differentiation matrix from (23). In (24) we didn't take boundary and matching conditions into account yet. For finding the eigenvalues, MATLAB's command `eig` is a very powerful tool.

Note, that by calling `eig`, finite set of eigenvalues is returned. However, the accuracy decreases with the number of the eigenvalue. Quantum graphs have infinitely many eigenvalues and in order to get many one needs to consider large N.

It might happen, that one is interested in merely few first eigenvalues or would be working with extensive matrices where the computations are not in the power of modern computers. Iteration methods may solve the problem. One of the most frequently used techniques for computing a few dominant eigenvalues is the Lanczos algorithm [16].

[2] The spectral Chebyshev matrix is also included in the MATLAB's Matrix Computation Toolbox under `chebspec` label.

Another argument for not computing all the eigenvalues is the fact, that in the case of the Laplacian on a graph with rationally dependent lengths of the edges, the eigenvalues are repeating within certain period (as proved in Proposition 2.3).

However, in our case we work with adequately small matrices and smooth solutions, so the `eig` command is more convenient for our purpose. Lanczos implementation might be one of the extensions to this thesis.

4.6 More about boundary conditions

First, let us demonstrate the boundary condition implementation on the interval. In the literature, the authors mainly concern about the Dirichlet boundary conditions $u(\pm 1) = 0$ since they are easy to implement. Indeed, according to [18] this may be achieved by omitting the outer rows and columns of the differentiation matrix and adjusting the length of vector u by setting $u_0 = u_N = 0$.

Different situation occurs when we take into account other types of boundary conditions. In general, there are two different approaches to implement boundary conditions for spectral methods:

1. restrict the set of interpolants to those, that satisfy the boundary conditions

2. do not restrict, but add additional equations to enforce the boundary conditions.

The first method is based on changing the form of interpolant basis to the so-called *boundary adapted* form. For theoretical background read the paper by Huang and Sloan [10], where the general form of the interpolant is provided. This method has been incorporated into the program `cheb2bc.m` by Weideman and Reddy, for more information see [19].

The second method is more flexible and suitable for more complicated problems. It is related to the so-called *tau methods* that appear in the field of Galerkin spectral methods. For more information see [3]. To present the idea behind the method, we apply it for the eigenvalue problem with Neumann boundary conditions (8).

Let us recall the differentiation matrix of second order (23). Then the discretized problem yields

$$\begin{cases} -D_N^{(2)} u = \lambda u, \\ u_0' = u_N' = 0, \end{cases}$$

where $u_i' = (D_N^{(1)} u)_i$ is the i-th element of the differentiated vector u. In other words, we impose $N-1$ equations using the second order differentiation matrix and 2 equations using the first order differentiation matrix. So we will end up solving $(N+1)\times(N+1)$ linear system of equations where $N-1$ equation enforce the condition $-u'' = \lambda u$ at the interior grid and 2 equations enforce $u' = 0$ at the outer grid points:

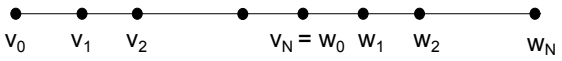

Figure 7: The grid for two connected intervals.

$$-Au = \lambda Bu, \tag{25}$$

in which A is the matrix build up mainly from $D_N^{(2)}$ by adding the outer rows coming from the boundary conditions. The most natural way to proceed is to replace the first row from $D_N^{(2)}$ with the first row from $D_N^{(1)}$, respectively last row from $D_N^{(2)}$ with last row from $D_N^{(1)}$. The matrix B is singular

$$B = \begin{pmatrix} 0 & 0 & 0 & \dots & 0 & 0 \\ 0 & 1 & 0 & \dots & 0 & 0 \\ 0 & 0 & 1 & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & 1 & 0 \\ 0 & 0 & 0 & \dots & 0 & 0 \end{pmatrix},$$

thus this is a generalized eigenvalue problem, which may be solved by the MATLAB command `eig(A,B)`.

4.7 Matching conditions

In the preceding subsection, we demonstrated how Dirichlet or Neumann boundary conditions can be introduced on a single interval. However, in more extensive cases we wish to impose matching conditions as well. This method is based on the same pattern, but needs one to be more careful with the signs and row/columns position.

To present the idea behind imposing matching conditions via spectral methods we first consider few explicit examples. Let us start with two intervals connected at one point satisfying the standard matching conditions.

4.7.1 Connected intervals

Two intervals are glued together (Figure 7) by imposing the standard matching conditions, i. e. Neumann boundary conditions at the endpoints. Mathematically, we are solving the discretized problem (following the formalism introduced in (1)):

$$\begin{cases} -D_N^{(2)} v = \lambda v, \\ -D_N^{(2)} w = \lambda w, \\ v_0' = w_N' = 0, \\ v_N = w_0, \\ v_N' = w_0', \end{cases} \tag{26}$$

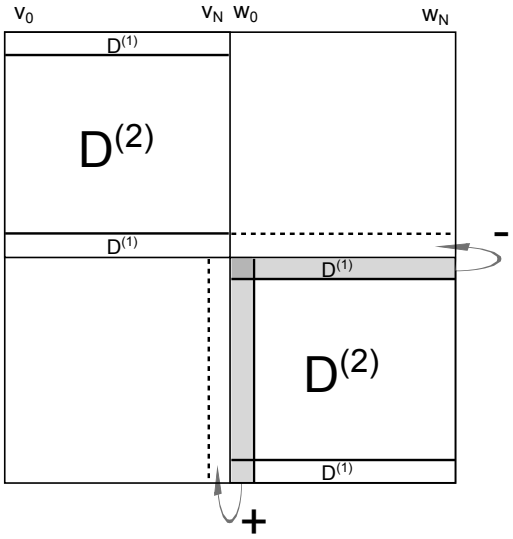

Figure 8: Differentiation matrix pattern for two connected intervals.

where $v = [v_0, \ldots, v_N]^T$ and $w = [w_0, \ldots w_N]^T$. The Laplacian matrix of the whole system (before implementing the matching conditions) is now block diagonal matrix of the size $2(N+1) \times 2(N+1)$. We aim to get the linear problem in the generalized form (25).

The procedure is depicted in Figure 8. First of all, we start with the conditions in (26) made up from derivatives. Utilizing v_0', v_N', w_0', w_N' means replacing all the outer rows in both $D_N^{(2)}$'s by the respective first order rows. For our nodes numbering, the Neumann boundary conditions also require setting right hand side matrix B having zero entries in the first and last position.

The derivative continuity condition required by the matching conditions (last row in (26)) may be enforced by *shifting the first row in the second block*, corresponding to the derivative in w_0, to the *last position of the first block*, corresponding to v_N', but with *minus sign*. Then, the Laplacian matrix A size reduces by one to $(2N+1) \times (2N+2)$ and the matrix B gains another zero on the $(N+1)$st position, i. e. the position where the condition is placed.

Similarly, the continuity of the function (second condition in (26)) may be taken into account by identifying v_n and w_0 which results into shifting the first *column* of the second block, corresponding to w_0, on the last position of the first block, corresponding to v_N. The obtained matrix is of the size $(2N+1) \times (2N+1)$. Right-hand side matrix B becomes identity matrix of the same size with the positions $(0,0), (N+1, N+1)$ and $(2N+1, 2N+1)$ excluded and replaced by zeros.

Note, that we always end up with a square matrix A. This was necessary since `eig` command requires square matrices, and the matrix B must be of the same size. Due to the singularity of B, some infinite eigenvalues may be

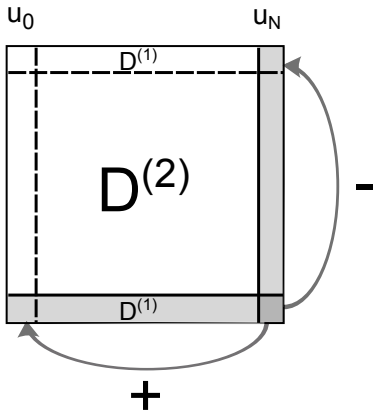

Figure 9: Differentiation matrix pattern for a ring graph.

obtained.

4.7.2 Loop graph

Next, we move to the ring graph. Here, the discretized equations obey

$$\begin{cases} -D_N^{(2)} u = \lambda u, \\ u_0 = u_N, \\ u_0' = u_N', \end{cases} \tag{27}$$

where $u = [u_0, \ldots, u_N]^T$. Let us follow the same procedure as in the previous case. Successively, introduce the first order differentiation rows in all nodes involved in the matching conditions in (27), namely in the first and last row. Then subtract those rows to enforce the derivative continuity condition $u_0' - u_N' = 0$ and set B equal to zero on the respective row.

To enforce the continuity condition we identify u_0 and u_N, i. e. the first and last columns should be added and placed on the first position.

The approach is graphically depicted in Figure 9. Observe, that the shifted rows and columns have always the same indexing. After applying the boundary and matching conditions, the Laplacian matrix shrinks from $(N+1)\times(N+1)$ to the size $N \times N$. Note, that the size drops always by the number of edges included minus number of pairs of matching conditions (which need to be stored).

4.7.3 Star graph

Another known case consisting of more than two edges is the star graph. For simplicity reasons, we consider three connected edges of the same length, but the process may be generalized for arbitrary number of edges meeting at the central point.

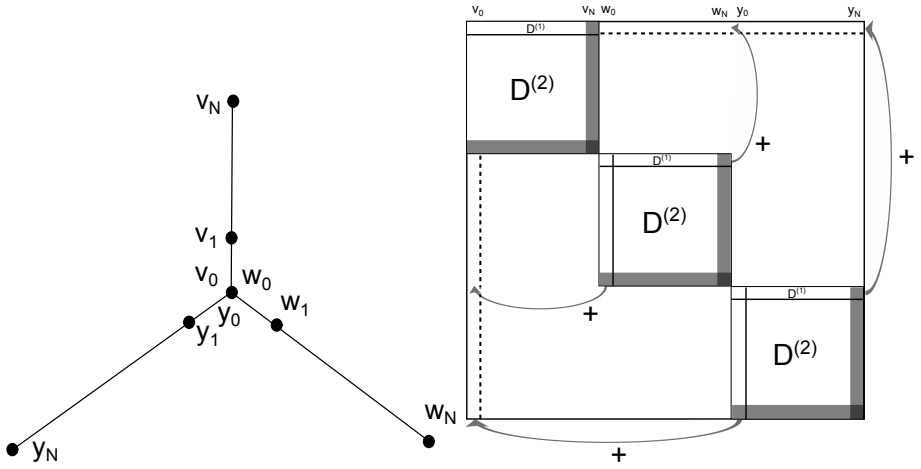

Figure 10: Star graph grid discretization and graphical Laplacian.

The discretized problem gives

$$\begin{cases} -D_N^{(2)} v = \lambda v, \\ -D_N^{(2)} w = \lambda w, \\ -D_N^{(2)} y = \lambda y, \\ v_0 = w_0 = y_0, \\ v_0' + w_0' + y_0' = 0, \\ v_N = w_N = y_N = 0, \end{cases} \tag{28}$$

where $v = [v_0, \ldots, v_N]^T$, $w = [w_0, \ldots, w_N]^T$ and $y = [y_0, \ldots, y_N]^T$. In Figure 10 there is the mesh as well as the graphical process of building the Laplacian matrix of the system of equations (28). As mentioned above, the rows enforce derivation continuity and columns the function continuity. Dirichlet boundary conditions are implemented by cutting off the respective rows and columns. One must not forget to provide zeros on the proper right-hand side matrix positions.

The final matrix gains the size $(3N-2) \times (3N-2)$ where five rows and columns has been omitted, namely 3 for the Dirichlet boundary conditions at each edge and 2 for number of edges meeting at one point minus one (one row has to be kept to store the matching condition equation). The MATLAB command `eig` computes not only eigenvalues but the eigenvectors as well. Some of them are shown in Figure 11.

4.7.4 Generalization

This process may be generalized for arbitrary graphs formed by n finite number of edges. Use the following algorithm:

1. Divide the vector u representing the eigenfunction on the graph into several vectors u_n representing the function on each edge.

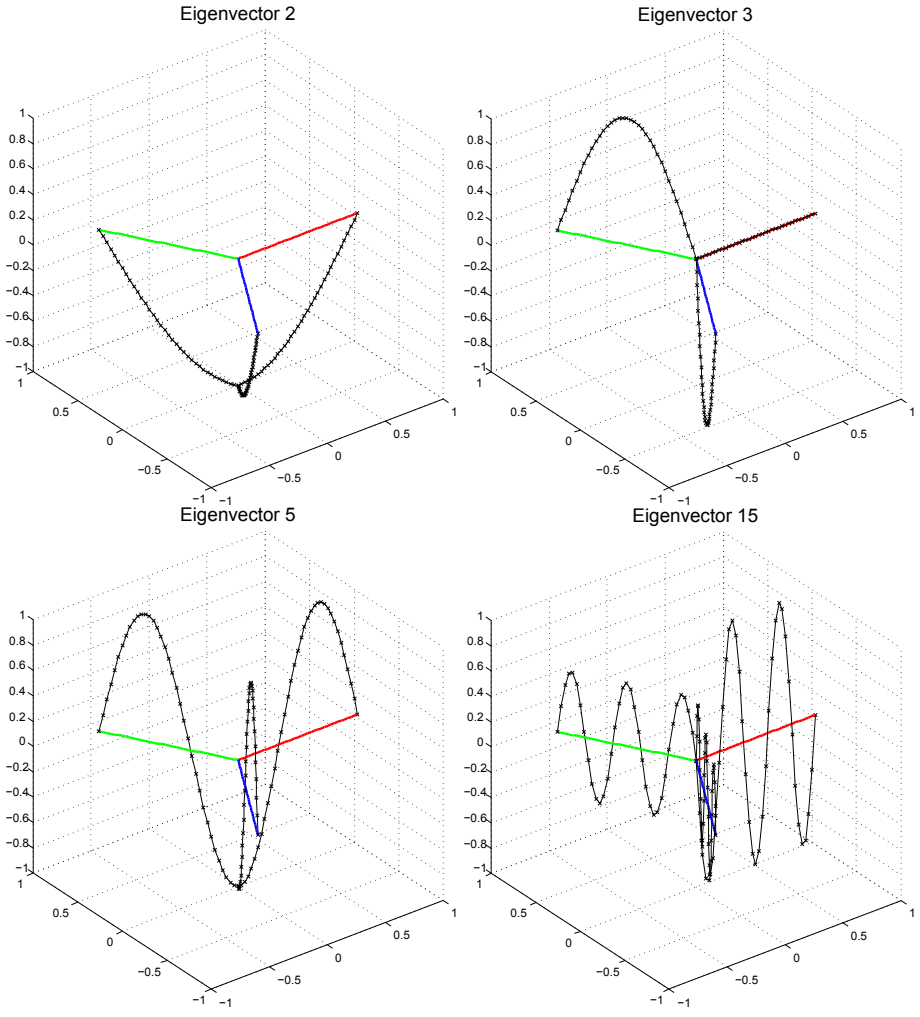

Figure 11: Some of the star graph eigenvectors.

2. Introduce block diagonal Laplacian matrix with the second order differentiation matrices $D_N^{(2)}$ on the diagonal, for each edge.

3. Mark all endpoints included in the matching and boundary conditions equations, the rows and columns corresponding to points with Dirichlet condition are to be erased.

4. Rows corresponding to Neumann boundary points or to matching points are replaced with respective first order differentiation rows.

5. Matching conditions are enforced by adding/subtracting the corresponding rows (watch the sign) and by adding columns respective to the edges meeting at each point.

6. Erase respective rows of the right-hand side matrix B to set boundary and matching condition to zero.

7. Employ generalized MATLAB's `eig` to get the eigenvectors and eigenvalues.

4.8 Adding potentials

As for implementation of the electric potential q in (3) we modify the discretized problem (24) to

$$-D_N^{(2)}u + Qu = \lambda u, \quad u = [u_0, \dots u_N]^T, \tag{29}$$

where Q is $(N+1)\times(N+1)$ diagonal matrix

$$Q = \begin{pmatrix} q_0 & 0 & \dots & 0 \\ 0 & q_1 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & q_N \end{pmatrix},$$

and q_i is the potential q evaluated in the Chebyshev node x_i (18):

$$q_i := q(x_i).$$

Similarly, for k edges, potential matrix Q would be diagonal of the size $k(N+1)\times k(N+1)$ with $q_0^1, q_1^1 \dots, q_N^1, q_0^2, \dots q_N^2, \dots, q_N^k$ on the diagonal, where q_i^j is q evaluated the ith node on jth edge. The computation is straightforward and is carried out in `qg2eigG.m`.

Magnetic potential a in the Schrödinger operator (2) is treated in `qg2eigG.m` as well. Let us introduce the unitary transformation

$$U_a|_{E_n} : u(x)|_{E_n} \mapsto \exp\left(-i\int_{x_{2n-1}}^{x} a(y)dy\right) u(x)|_{E_n}, \tag{30}$$

which transforms the magnetic Schrödinger operator to

$$L_{q,a} = U_a^{-1} L_{q,0} U_a.$$

Defining $\tilde{u}(x)|_{E_n} = \exp\left(-i\int_{x_{2n-1}}^{x} a(y)dy\right) u(x)|_{E_n}$ we can rewrite the endpoint terms as

$$\tilde{u}(x_{2n-1}) = u(x_{2n-1}), \quad \tilde{u}(x_{2n}) = e^{-i\varphi_n} u(x_{2n}), \tag{31}$$

where φ_n denotes $\varphi_n := \int_{x_{2n-1}}^{x_{2n}} a(y)dy$.

The extension of the discrete problem (29) thus yields

$$-D_N^{(2)}\tilde{u} + Q\tilde{u} = \lambda\tilde{u}, \quad \tilde{u} = [\tilde{u}_0, \dots \tilde{u}_N]^T,$$

where the endpoint values in the matching conditions are taken in the form (31).

4.9 Implementation

We implemented this process in the MATLAB functions that could be freely downloaded in http://gemma.ujf.cas.cz/∼malenova/download.html. There exist more versions of the algorithm. Successively:

- `qg2eig.m`- the Dirichlet boundary conditions at all the boundary points are considered. All the internal edges satisfy the standard matching conditions.
- `qg2eigN.m` implements standard matching conditions for Laplace operator.
- `qg2eigG.m` is a general version implementing standard magnetic Schrödinger operator (2). It adds electric and magnetic potentials by modifying the diagonal entries (29) and introducing unitary transformation (30).

Observe that the graph does not need to be directed, however, for the computational reasons every edge has a starting and an ending point. The solution does not depend on this convention, but it is crucial to keep it unchanged during the computation to correctly utilize the normal derivative direction defined in (1).

In our algorithm, *incidence matrix* in the input is required. More precisely, it is an $n \times m$ matrix I, where n stands for the number of edges and m for the number of points, defined as:

$$I_{ij} = \begin{cases} -1 & \text{if } j \text{ is the starting point of edge i,} \\ 1 & \text{if } j \text{ is the end point of edge i,} \\ 2 & \text{if } i \text{ is the loop edge at } j-\text{th point,} \\ 0 & \text{otherwise.} \end{cases}$$

As a complement for the differentiation matrices build-up we need already mentioned function `chebdif.m` by Weideman & Reddy, which is available on the webpage http://dip.sun.ac.za/∼weideman/research/differ.html. The documentation to the function is provided in [19].

Another function required for running the `qg2eig` is `v2g.m`. This function assembles a vector and a matrix in the input, providing the matrix non-zero

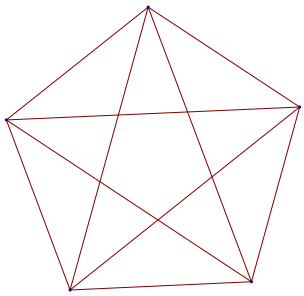

Figure 12: Complete graph with 5 nodes denoted by K_5.

elements will be replaced by the vector entries. Obviously, the number of non-zero entries and the vector length has to be the same. In the output, we obtain matrix of the same size.

As the output of `qg2eig` we get the eigenvectors and eigenvalues corresponding to the general quantum graph defined by the incidence matrix. The program is able to handle loops and non-connected components and works very fast even for larger graphs. All the edge lengths are set to 2.

Recently, Chebyshev polynomials-based computations gained certain popularity, especially due to the Oxford University project *Chebfun* supervised by Lloyd Nick Trefethen that goes back to 2002 [1]. Chebfun is an open source package extending MATLAB environment providing one may conduct the computations in *functional form* (instead of vectorized form). As a new feature of Chebfun, the algorithm for computing the spectra of quantum graphs described above is going to be implemented as well. The current version of Chebfun is available at http://www2.maths.ox.ac.uk/chebfun/.

5 Applications

Once we have a program computing the spectrum of a magnetic Schrödinger operator on a general graph, it is handy to draw our attention on inverse problems. Is the spectrum carrying any information about the properties of the graph? We investigated some interesting issues including trace formula and spectral gap.

5.1 Trace formula

For planar graphs, the *Euler characteristics* χ is given by

$$\chi = M - N, \tag{32}$$

where M, N is number of vertices and edges respectively. Trees have $\chi = 1$, whereas all the other graphs have $\chi \leq 0$.

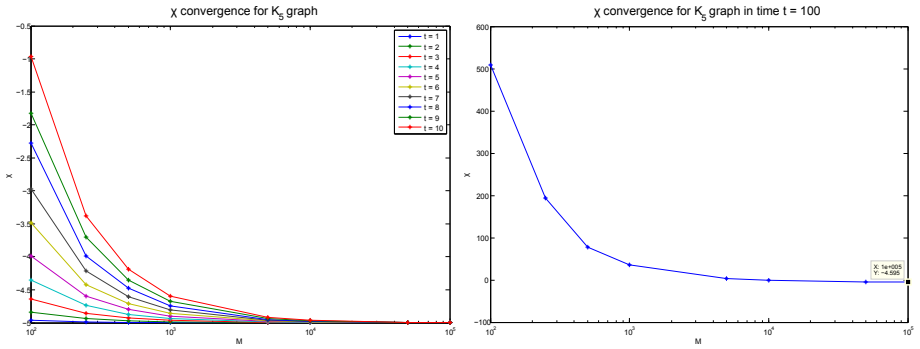

Figure 13: Euler characteristics χ for K_5 graph (Figure (12)) at time $t = 1$. On the x-axis, there is number of terms used.

As a function of a quantum graph's spectrum, Euler characteristics χ also comes out from a trace formula [14] as a byproduct:

$$\chi = 2m_s(0) + 2 \lim_{t\to\infty} \sum_{k_n \neq 0} \cos(k_n/t) \left(\frac{\sin(k_n/2t)}{k_n/2t} \right)^2, \tag{33}$$

where $m_s(0)$ is spectral multiplicity of eigenvalue 0 and k_n^2 is n-th eigenvalue. Limit and sum may not be exchanged for the expression not to diverge (as $\cos(k_n/t) \left(\frac{\sin(k_n/2t)}{k_n/2t} \right)^2 \longrightarrow 1$ providing $t \to \infty$).

Let us define the residual

$$R_N := \sum_{n=N}^{\infty} \cos(k_n/t) \left(\frac{\sin(k_n/2t)}{k_n/2t} \right)^2.$$

A rough estimate is made utilizing the upper bound for goniometric functions and Weyl's asymptotics $k_n \sim \frac{\pi n}{L}$, where L is the total length of the graph:

$$|R_N| = \left| \sum_{n=N}^{\infty} \underbrace{\cos\left(\frac{\pi n}{Lt}\right)}_{\leq 1} \underbrace{\sin^2\left(\frac{\pi n}{2Lt}\right)}_{\leq 1} \left(\frac{2Lt}{\pi n}\right)^2 \right| \leq \frac{4L^2t^2}{\pi^2} \left| \sum_{N}^{\infty} \frac{1}{n^2} \right| \leq$$

$$\leq \frac{4L^2t^2}{\pi^2} \int_{N-1}^{\infty} \frac{1}{x^2}\, dx = \frac{4L^2t^2}{\pi^2} \frac{1}{N-1}. \tag{34}$$

Example. Say we want to compute Euler characteristics of a K_5 graph (Figure 12) using the formula (33) achieving precision $|R_N| < 0.25$. The estimate (34) states, that one would need to take approximately $N = 650$ terms at time $t = 1$ to meet the requirements.

As to theoretical estimation, the number of terms necessary for the numerical computation is far lower. According to (32), the correct answer is $\chi = 5 - 10 =$

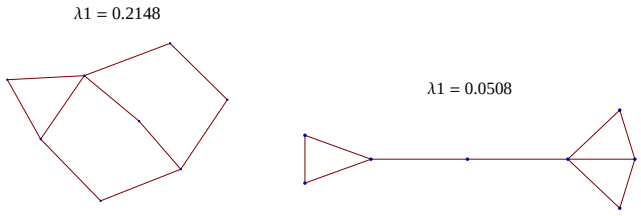

Figure 14: The higher is the spectral gap, the more is the graph prone to be robust. In this example of two graph with the same total lengths and vertex valencies (with standard Laplace operator).

-5. We may take use of Proposition 2.3 and compute only the first period which makes the process much more precise and faster, since we need large amount of eigenvalues. In Figure 13 one may observe the convergence of χ for various times. In case $t = 1$ the curve needs barely $N = 20$ terms to achieve the same precision as required.

5.2 Spectral gap

In the literature, spectral gap[3] was previously investigated for discrete graphs and refers to the second eigenvalue of the Laplacian (on a connected graph). It may apply to discrete as well as to continuous graphs.

Recently, the spectral gap was also investigated numerically on large random graphs [9]. The research concluded, vaguely said, that λ_1 may be taken as a measure of *synchronizability and robustness*. It was shown, that the higher is the spectral gap, the more is the graph prone to be robust and synchronic. This found its application in neuron networks or signal transfer area.

5.2.1 Synchronizability

Synchronizability describes the ability of a system to cooperate and synchronize the phase. This ability is highly appreciated for large systems of items that are required to transfer a signal effectively without losing information or being sensitive to perturbations that may cause damage on the data. This is for example the case of neuron system.

5.2.2 Robustness

Robustness is the ability of the system to cope with errors. This property may be crucial for example for signal transferring or connected networks ranging from studies of internet to army supply strategies.

One can demonstrate the robustness in the example shown in Figure 14. Those two metric graphs have both the same number of edges, length and

[3]It is sometimes called algebraic connectivity or Fiedler value, according to Czechoslovak mathematician, who first described the algebraic connectivity in [7].

sequence of vertex valencies. However, the one on the left-hand side has considerably higher spectral gap (for standard Laplace operator). At the same time, the graph on the left is more robust in the sense that destroying one of the vertex (i. e. router, army base) or an edge (i. e. wires, logistic path) does not necessarily mean destroying the whole connection or cutting off to separate parts. On the other hand, the same damage would be more likely devastating for the other network on the right, what is reflected in lower λ_1.

Let us first start with proving some statements about discrete graphs. In spite they in general do not show the same properties as continuous graphs, some degree of coherence has been observed.

5.3 Discrete graphs

The Laplacian on discrete graphs is defined as follows:

$$L = V - A, \tag{35}$$

where A is the adjacency matrix and V vertex diagonal matrix respectively:

$$A = \begin{cases} 1 & \text{if the vertices } i \text{ and } j \text{ are connected,} \\ 0 & \text{otherwise,} \end{cases} \qquad V = \operatorname{diag}(v_1, v_2, \ldots, v_n),$$

where v_i is the valence of the vertex i. The original publication by Fiedler [7] devoted to spectral gap provides many relations and estimates regarding the second eigenvalue.

Spectral gap is a monotonous function of the set of edges, in other words, cutting off an edge always causes drop of the second eigenvalue or keeps it unchanged, provided we have the same set of vertices.

Proposition 5.1. *Let G' be a discrete graph obtained from G by adding one edge connecting vertices n_1 and n_2 and L the discrete Laplacian defined by* (35)*. Then the following holds:*

1. *λ_0 is the lowest eigenvalue for both $L(G)$ and $L(G')$. The corresponding eigenfunction is $\psi = \hat{1}$, where $\hat{1}$ denotes the vector (of appropriate size) built up from ones.*

2. *The second eigenvalues satisfy the following identity:*

$$\lambda_1(G) \leq \lambda_1(G').$$

3. *The equality $\lambda_1(G) = \lambda_1(G')$ holds if and only if the second eigenfunction ψ_1^G on the G graph may be chosen attaining equal values at the vertices n_1 and n_2*

$$\psi_1^G(n_1) = \psi_1^G(n_2).$$

Proof. The first statement is a standard result and can be proven by substituting $\psi_0 = \hat{1}$ into the equation $L\psi_0 = 0$.

The second statement follows from the fact that

$$L(G') - L(G) = \begin{pmatrix} & \vdots & & \vdots & \\ \dots & 1 & \dots & -1 & \dots \\ & \vdots & & \vdots & \\ \dots & -1 & \dots & 1 & \dots \\ & \vdots & & \vdots & \end{pmatrix} \tag{36}$$

is a matrix with just four non-zero entries. It is easy to see that the matrix is positive semi-definite, since the eigenvalues are 0 (with multiplicity $n-1$ where n is the number of nodes) and 2 (simple eigenvalue) and therefore $L(G')-L(G) \geq 0$ which implies the statement.

To prove the last assertion, let us remember that $\lambda_1(G')$ can be calculated using

$$\lambda_1(G') = \min_{\psi \perp 1} \frac{\langle \psi, L(G')\psi \rangle}{\langle \psi, \psi \rangle} \geq \min_{\psi \perp 1} \frac{\langle \psi, L(G)\psi \rangle}{\langle \psi, \psi \rangle}$$

We have equality in the last formula if and only if ψ minimizing the first quotient is such that $(L(G') - L(G))\psi = 0$, i. e. $\psi(n_1) = \psi(n_2)$.

□

Next, we are interested what happens if we add a pending edge, i. e. an edge connected to the graph in one already existing node.

Proposition 5.2. *Let G be a connected discrete graph and let G' be another graph obtained from G by adding one vertex and one edge connecting the new vertex with the vertex n_1. Then the following holds:*

1. *The second eigenvalues satisfy the following inequality:*

$$\lambda_1(G) \geq \lambda_1(G').$$

2. *The equality $\lambda_1(G) = \lambda_1(G')$ holds if and only if every second eigenfunction ψ_1^G is equal to zero at n_1*

$$\psi_1^G(n_1) = 0.$$

Proof. Let us define the following vector on G':

$$\varphi(n) := \begin{cases} \psi_1^G(n), & \text{on } G, \\ \psi_1^G(n_1) & \text{on } G' \backslash G. \end{cases}$$

This vector is not orthogonal to the zero energy eigenfunction $\hat{1}$. Therefore consider vector γ shifted by a constant c

$$\gamma(n) := \varphi(n) + c,$$

where c is chosen so that the orthogonality condition in the finite-dimensional space $l_2(G')$ holds

$$0 = \langle \gamma, \hat{1} \rangle_{l_2(G')} = \langle \psi_1^G, \hat{1} \rangle_{l_2(G)} + \psi_1^G(n) + cm,$$

where m is the number of edges of G'. This implies

$$c = -\frac{\psi_1^G(n_1)}{m}.$$

Using this vector the following estimate on the second eigenvalue may be obtained:

$$\lambda_1(G') \leq \frac{\langle L\gamma, \gamma \rangle_{l_2(G')}}{\|\gamma\|^2_{l_2(G')}} = \frac{\langle L\psi_1^G, \psi_1^G \rangle_{l_2(G)}}{\|\psi_1^G\|^2_{l_2(G)} + c^2 m + |\psi_1^G(n_1) + c|^2} \leq \lambda_1(G).$$

The last inequality follows from the fact that

$$\langle L\psi_1^G, \psi_1^G \rangle_{l_2(G)} = \lambda_1(G) \|\psi_1^G\|^2,$$

and

$$\|\psi_1^G\|^2_{l_2(G)} + c^2 m + |\psi_1^G(n_1) + c|^2 \geq \|\psi_1^G\|^2_{l_2(G)}.$$

Note, that we have equality if and only if $c = 0$ and $|\psi_1^G(n_1) + c|^2 = 0$ which implies $\psi_1^G(n_1) = 0$. □

5.4 Continuous graphs

As was already mentioned, spectral gap was extensively investigated for discrete graphs. We want to explore similar concept for continuous quantum graphs as well. Satisfying additional conditions, we may prove an analog to Proposition 5.1 and Proposition 5.2 respectively.

Theorem 5.3. *Let Γ be a connected metric graph and L^{st} the corresponding standard Laplace operator (Definition 3). Let Γ' be a metric graph obtained from Γ by adding an edge of length l between the vertices m_1 and m_2. Then the following holds:*

1. *$\lambda_0 = 0$ is the lowest eigenvalue for $L^{st}(\Gamma)$ and $L^{st}(\Gamma')$ with the eigenfunctions $u_0 \equiv 1 \in L_2(\Gamma)$ and $u_0' \equiv 1 \in L_2(\Gamma')$.*

2. *Assume that the eigenfunction u_1 corresponding to the second eigenvalue can be chosen such that*

$$u_1(m_1) = u_1(m_2).$$

Then the following inequality for second eigenvalues hold:

$$\lambda_1(\Gamma) \geq \lambda_1(\Gamma').$$

Proof. The proof of the first statement is standard and is carried out in Proposition 2.2.

To prove the second inequality, let us consider the second eigenfunction $u_1(\Gamma)$ for $L^{st}(\Gamma)$. We introduce the function

$$f(x) = \begin{cases} u_1(x), & x \in \Gamma, \\ u_1(m_1)(= u_1(m_2)) & x \in \Gamma' \backslash \Gamma. \end{cases}$$

This function is not orthogonal to the zero energy eigenfunction. Let us adjust the constant c so that $g(x) = f(x) + c$ is orthogonal to 1 in $L_2(\Gamma')$:

$$0 = \langle g(x), 1 \rangle_{L_2(\Gamma')} = \underbrace{\langle u_1(x), 1 \rangle_{L_2(\Gamma)}}_{=0} + u_1(m_1) l + c\mathcal{L}' = 0,$$

where $\mathcal{L}'$ is the total length of the graph Γ'. This implies

$$c = -\frac{u_1(m_1) l}{\mathcal{L}'}.$$

Now we are ready to get an estimate for $\lambda_1(\Gamma')$ using

$$\lambda_1(\Gamma') \leq \frac{\langle L^{st}(\Gamma') g, g \rangle_{L_2(\Gamma')}}{\|g\|^2_{L_2(\Gamma')}}.$$

The numerator yields

$$\langle L^{st}(\Gamma') g, g \rangle_{L_2(\Gamma')} = \langle L^{st}(\Gamma) u_1, u_1 \rangle_{L_2(\Gamma)} = \lambda_1(\Gamma) \|u_1\|^2_{L_2(\Gamma)},$$

and the denominator is

$$\begin{aligned} \|g\|^2_{L_2(\Gamma')} &= \|u_1 + c\|^2_{L_2(\Gamma)} + |u_1(m_1) + c|^2 l = \\ &= \|u_1\|^2_{L_2(\Gamma)} + c^2 \mathcal{L} + |u_1(m_1) + c|^2 l \geq \\ &\geq \|u_1\|^2_{L_2(\Gamma)} \end{aligned}$$

It follows, that

$$\lambda_1(\Gamma) \geq \lambda_1(\Gamma').$$

□

Theorem 5.4. *Let Γ be a connected metric graph and let Γ' be another graph obtained from Γ by adding one vertex and one edge of length l connecting the new vertex with the vertex m_1. The standard Laplace operator $L^{st}(\Gamma)$, respectively $L^{st}(\Gamma')$, is acting on edges. Then the second eigenvalues satisfy the following inequality:*

$$\lambda_1(\Gamma) \geq \lambda_1(\Gamma').$$

Proof. Let us define the following function on Γ':

$$f(x) := \begin{cases} u_1(x), & x \in \Gamma, \\ u_1(m_1) & x \in \Gamma' \backslash \Gamma. \end{cases}$$

This function is in general not orthogonal to the zero energy eigenfunction $1 \in L_2(\Gamma')$. Therefore consider function g shifted about a constant c

$$g(x) := f(x) + c,$$

where c is chosen so that the orthogonality condition in $L_2(\Gamma')$ holds

$$0 = \langle g(x), 1 \rangle_{L_2(\Gamma')} = \underbrace{\langle u_1, 1 \rangle_{L_2(\Gamma)}}_{=0} + u_1(m_1) l + c\mathcal{L}',$$

where $\mathcal{L}'$ is the total length of Γ'. This implies

$$c = -\frac{u_1(m_1) l}{\mathcal{L}'}.$$

Using this vector the following estimate on the second eigenvalue may be obtained:

$$\lambda_1(\Gamma') \le \frac{\langle L^{st} g, g \rangle_{L_2(\Gamma')}}{\|g\|^2_{L_2(\Gamma')}} = \frac{\langle L^{st} u_1, u_1 \rangle_{L_2(\Gamma)}}{\|u_1\|^2_{L_2(\Gamma)} + c^2 \mathcal{L} + |u_1(m_1) + c|^2 l} \le \lambda_1(\Gamma).$$

The last inequality follows from the fact that

$$\langle L^{st} u_1, u_1 \rangle_{L_2(\Gamma)} = \lambda_1(\Gamma) \|u_1\|^2,$$

and

$$\|u_1^\Gamma\|^2_{L_2(\Gamma)} + c^2 \mathcal{L} + |u_1(m_1) + c|^2 l \ge \|u_1\|^2_{L_2(\Gamma)}.$$

□

To demonstrate the conclusions of the above theorems in applications, we consider several examples.

Example. Consider graph Γ' consisting of two intervals of lengths a and b connected in parallel. This graph is equivalent to the circle of length $a + b$. The spectrum of the Laplacian $L(\Gamma')$ is (according to (10)):

$$\sigma(L(\Gamma')) = \left\{ \left(\frac{2\pi}{a+b} \right)^2 n^2 \right\}_{n=0}^{\infty},$$

where all the eigenvalues except for zero have double multiplicity.

Deleting the edge of length b we get Γ formed by single edge of length a. The spectrum of $L(\Gamma)$ is

$$\sigma(L(\Gamma)) = \left\{ \left(\frac{\pi}{a} \right)^2 n^2 \right\}_{n=0}^{\infty},$$

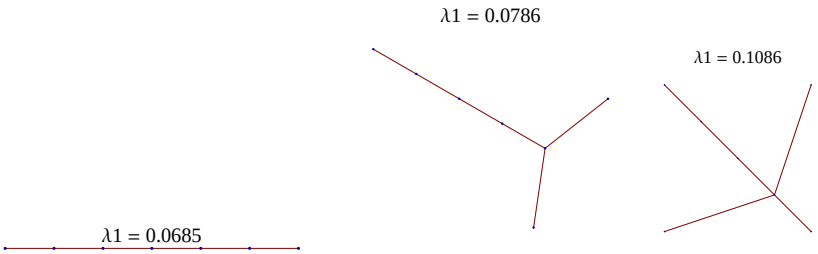

Figure 15: Equilateral graphs of the same total length. One can see that the more branching, the higher is the spectral gap.

as shown in (9), this time with multiplicity one. Thus we have $\lambda_1(\Gamma') = \left(\frac{2\pi}{a+b}\right)^2$ and $\lambda_1(\Gamma) = \left(\frac{\pi}{a}\right)^2$. Any relation between these values is possible:

$$b > a \;\Rightarrow\; \lambda_1(\Gamma') < \lambda_1(\Gamma),$$
$$b < a \;\Rightarrow\; \lambda_1(\Gamma') > \lambda_1(\Gamma).$$

Example. Consider the graph Γ'' formed by three parallel edges of lengths a, b and c and the graph Γ' formed by three parallel edges of lengths a and b. The second eigenfunction for $L(\Gamma')$ can always be chosen so that $u_1(m_1) = u_1(m_2)$. Then, in accordance to Theorem 5.3, the second eigenvalue for $L(\Gamma'')$ is less or equal to the second eigenvalue for $L(\Gamma')$:

$$\lambda_1(\Gamma'') \le \lambda_1(\Gamma').$$

5.5 Rayleigh theorem for quantum graphs

The classical Rayleigh theorem states that the gap between the lowest two eigenvalues of the Neumann Laplacian in a domain of fixed area is maximal if the domain is circle. Our aim is to prove an analog of this theorem for Laplace operators on graphs with standard matching conditions at the vertices. This section is a content of the article [15] that is currently in preparation.

Motivation

Let us consider a single string of the length $\mathcal{L}$. The second eigenvalue is well-known: $\lambda_1 = \frac{\pi^2}{\mathcal{L}^2}$ (see (9)). Now, take a graph of the same length built up of a string with a simple *branching*, i. e. including a vertex having valency at least 3, see Figure 15. It has been numerically computed that such branching causes increase of the second eigenvalue. Indeed, after splitting the string to two branches, the spectral gap increases from $\lambda_1 = 0.0685$ to $\lambda_1 = 0.0786$, three branches brought further increase to $\lambda_1 = 0.1086$. This suggests a hypothesis, that simple string is the least robust and synchronizable of all graphs of the same total length, in other words, branching always causes increase in λ_1. This is the content of the following section.

Theorem 5.5. *Let Γ be a connected metric graph with the total length $\mathcal{L}$ and $L^{st}(\Gamma)$ the corresponding Laplace operator defined on the domain of functions satisfying standard matching conditions at the vertices. Consider as well the graph $\Delta_{\mathcal{L}}$ formed by one interval of length $\mathcal{L}$ and the corresponding standard Laplacian $L(\Delta)$. Point $\lambda_0 = 0$ is a simple eigenvalue for both Laplacians. Then the following inequality holds for the lowest nontrivial eigenvalues:*

$$\lambda_1(\Gamma) \geq \lambda_1(\Delta).$$

Proof. Consider the eigenfunction ψ_1 corresponding to the eigenvalue $\lambda_1(\Gamma)$. This eigenvalue is a minimum of the *Rayleigh quotient*

$$\lambda_1(\Gamma) = \min_{\varphi \perp 1} \frac{\int_\Gamma |\varphi'(x)|^2 \, dx}{\int_\Gamma |\varphi(x)|^2 \, dx}, \tag{37}$$

taken over all continuous functions on Γ orthogonal to the ground state $\psi_0 \equiv 1$. The minimum is attained for $\varphi = \psi_1$ and thus the eigenfunction is a minimizer of (37).

Consider the graph Γ^*- the double cover of Γ- obtained from Γ by doubling each edge. The new edges connect the same vertices. Any function from $L_2(\Gamma)$ can be lifted to $L_2(\Gamma^*)$ in a symmetric way by assigning it the same values on the pairs of edges as on the edges of the original graph Γ. The function ψ_1^* obtained in this way obviously satisfies

$$\lambda_1(\Gamma) = \frac{\int_{\Gamma^*} |\psi_1^{*\prime}(x)|^2 \, dx}{\int_{\Gamma^*} |\psi_1^*(x)|^2 \, dx}.$$

Every vertex in Γ^* has even valency and therefore there exists a closed (Eulerian) path $\mathcal{P}$ (see [4], [8]) on Γ^* crossing each edge precisely one time. The path $\mathcal{P}$ can be identified with a loop $S_{2\mathcal{L}}$ of length $2\mathcal{L}$.

Consider now the function ψ_1^* as a function on the loop $S_{2\mathcal{L}}$. It is continuous function orthogonal to the ground state function on the loop and therefore gives an upper estimate for the corresponding eigenvalue for the Laplacian on the loop

$$\lambda_1(S_{2\mathcal{L}}) \leq \frac{\int_{S_{2\mathcal{L}}} |\psi_1^{*\prime}(x)|^2 \, dx}{\int_{S_{2\mathcal{L}}} |\psi_1^*(x)|^2 \, dx} = \lambda_1(\Gamma).$$

We obtain the result noticing that

$$\lambda_1(S_{2\mathcal{L}}) = \lambda_1(\Delta_{\mathcal{L}}),$$

that follows from (9) and (10). □

Example. Comparing string and 3-star graph of the same total lengths $\mathcal{L}$ always gives higher λ_1 for the latter one. As shown above, the string's first eigenvalue is $\frac{\pi^2}{\mathcal{L}^2}$ while the λ_1 for star graph is given by (14), which may be computed numerically.

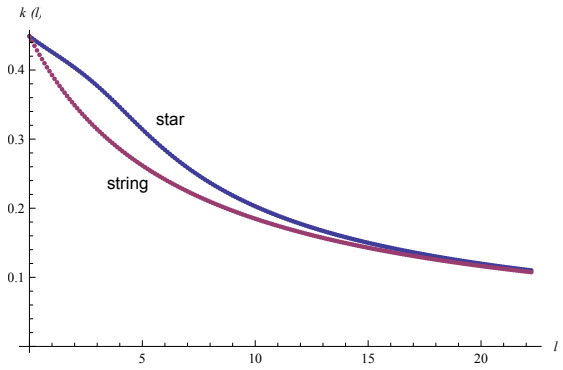

Figure 16: Comparison of the second eigenvalue for the star and string graph. One length l of the star graph is variable, the other two are fixed. λ_1 for star graph is always higher than for the string of the same length.

In Figure 16, the comparison is performed. There are two edges of the star graph fixed ($l_1 = 2, l_2 = 5$) and l_3 is variable and being marked on the x-axis. The blue plot represents the star while red line depicts string graph of the same total length, i. e. the eigenvalue $\frac{\pi^2}{(l_1+l_2+l_3)^2}$.

One can observe that the difference increases for small l_3. The length of the edges in the star graph are comparable in this regime. However, while increasing one of the lengths to infinity, the star graph begins to act more and more like a string and their spectra almost coincide.

Special case of two edges of the star graph having the same length was considered. The case $l_1 = l_3 = l$ was analytically resolved in Section 3.4.

The numerical solution is presented in Figure 17. We set the values to $l_1 = l_3 = 5$ and adjust the third star graph edge. In the first part, the eigenvalue follows the first type of solution given by (15), what changes to solution of (16) while crossing the point $l = 5$.

The first type of solution corresponds to the case when the eigenfunction on the edge e_2 stays constant. Whereas after crossing the equilateral state $l_1 = l_2 = l_3$ the solution becomes wavy on all the edges.

6 Conclusion

In the master thesis we were concerned about spectral properties of quantum graphs. As the spectra are not explicitly computable in general case, we provided a numerical tool capable to resolve the spectral problem for arbitrary compact equilateral graph providing we consider the standard Laplace operator (Def (3)). Subjected to some changes, we are able to add electric and magnetic potentials.

Having the spectrum was the starting point for exploring related properties of the quantum graphs. First, we computed the Euler characteristics (33) from the trace formula using far less eigenvalues than estimated. Finally, we concen-

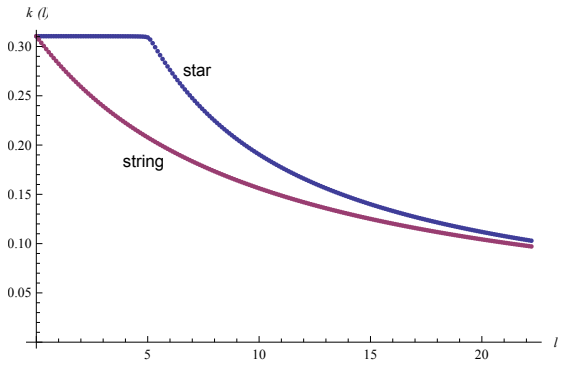

Figure 17: Comparison of the second eigenvalue for the star and string graph. The lengths are $l_1 = l_3 = 5$. One may observe, that crossing the length $l_3 = 5$ the eigenvalue switches to different type.

trated over the second eigenvalue λ_1, also called spectral gap, which has special significance as a measure of synchronizability and robustness. It has been shown that dropping one edge does not necessarily mean increase in λ_1, but that there is sufficient condition for achieving it. Next, we proved that the string has the lowest spectral gap among all graphs of the same total length.

This topic provides plenty possibilities for further extension. As for the numerical part, Chebfun open source package implementation is the next goal. Regarding the analytical part, some conjectures about the second eigenvalue of the standard Laplacian are to be proved, above all the fact that while string is the graph with the lowest λ_1, *complete graph* should be the one with the *highest* λ_1 among the graphs with the same total length and number of vertices.

References

[1] Z. Battles, L. N. Trefethen, *An Extension of MATLAB to Continuous Functions and Operators*, SIAM J. Sci. Comput., Vol. 25, No. 5, 2004, pp. **1743-1770**.

[2] C. Canuto, M. Y. Hussaini, A. Quarteroni, T. A. Zang, *Spectral Methods: Fundamentals in Single Domains*, Springer-Verlag, Berlin, Heidelberg, 2006.

[3] J. J. Dongarra, B. Straughan, D. W. Walker, *Chebyshev tau-QZ Algorithm Methods for Calculating Spectra of Hydrodynamic Stability Problems*, Applied Numerical Mathematics 22(4): **399-435**, 1996, ISSN 0168-9274.

[4] L. Euler, *Solutio problematis ad geometriam situs pertinentis*, *Comment. Academiae Sci. I. Petropolitanae* 8 (1736), **128-140**.

[5] P. Exner, O. Post, *Quantum Networks Modeled by Graphs*, Proceedings of the Joint Mathematics/Physics Workshop "Quantum Few-Body System" (Aarhus 2007), AIP Conf. Proc., vol. 998; Melville, NY, 2008, pp. **1-17**.

[6] P. Exner and P. Šeba, *Free quantum motion on a branching graph*, Rep. Math. Phys., 28 (1989), **7-26**.

[7] M. Fiedler, *Algebraic Connectivity of Graphs*, Czechoslovak Mathematical Journal, 23 (98), 1973, Praha, **298-305**.

[8] C. Hierholzer, Chr. Wiener, *Über die Möglichkeit, einen Linienzug ohne Wiederholung und ohne Unterbrechung zu umfahren*, *Math. Ann.* **6** (1873), no. 1, **30-32**.

[9] M. Holroyd, *Connectivity and Synchronizability of Discrete Complex Systems*, 2006, Williamsburg, VA, Honors thesis.

[10] W. Huang, D. M. Sloan, *The Pseudospectral Method for Third-Order Differential Equations*, SIAM Journal on Numerical Analysis, Vol. 29, 1992, **1626-1647**.

[11] V. Kostrykin, R. Schrader *Laplacians on Metric Graphs: Eigenvalues, Resolvents and Semigroups*, Quantum Graphs and Their Applications, Contemp. Math., Am. Math. Soc., Providence, 2006, pp. **201-225**.

[12] P. Kuchment, *Quantum Graphs I: Some Basic Structures*, Waves Random Media, 14, **107-128**, 2004.

[13] P. Kurasov, *Inverse Problems For Quantum Graphs: Recent Development and Perspectives*, Acta Physica Polonica A, 120 , v6A (2011), **132-141**.

[14] P. Kurasov, *Quantum Graphs: Spectral Theory and Inverse Problems*, in preparation.

[15] P. Kurasov and S. Naboko, On Rayleigh theorem for quantum graphs, in preparation.

[16] G. Meurant and Z. Strakos, *The Lanczos and conjugate gradient algorithms in finite precision arithmetic*, Acta Numerica, 15, Cambridge University Press, 2006, pp. **471-542**.

[17] K. Pankrashkin, *Spectra of Schrödinger Operators on Equilateral Quantum Graphs*, Letters in Math. Physics (2006) 77: **139-154**.

[18] L. N. Trefethen, *Spectral Methods in MATLAB*, SIAM, Philadelphia (2000).

[19] J. A. C. Weideman, S. C. Reddy, *A MATLAB Differentiation Matrix Suite*, ACM Transactions on Mathematical Software, Vol. 26, No. 4, 2000, **465-519**.