Acknowledgements

- Joint work with Mario Arioli (Berlin)
- Support: NSF, Emerson Center for Scientific Computation, Berlin Mathematical School, TU Berlin
The purpose of this talk is to introduce the audience to a class of mathematical models known as quantum graphs, and to describe some numerical methods for investigating such models.

Roughly speaking, a quantum graph is a collection of intervals glued together at the end-points (thus forming a metric graph) and a differential operator ("Hamiltonian") acting on functions defined on these intervals, coupled with suitable boundary conditions at the vertices.
Quantum graphs are becoming increasingly popular as mathematical models for a variety of physical systems including conjugated molecules (such as graphene), quantum wires, photonic crystals, carbon nanostructures, thin waveguides, etc.

Example: Graphene
Example: Carbon nanostructures
Example: Polystyrene
Other potential applications include the modeling of phenomena such as information flow, diffusion and wave propagation in complex networks (including social and financial networks), blood flow in the vascular network, electrical signal propagation in the nervous system, traffic flow simulation, and so forth.
Early structure of the Internet
While the theory of quantum graphs is very rich (well-posedness results, spectral theory, etc.), there is very little in the literature on the numerical analysis of such models.

In this lecture we consider numerical methods for the analysis of quantum graphs, focusing on simple model problems.

This is still very much work in progress!
A combinatorial graph $\Gamma$ is a pair $(\mathcal{V}, \mathcal{E})$ where $\mathcal{V} = \{v_j\}_{j=1}^{N}$ is a set of vertices, or nodes, and $\mathcal{E} = \{e_k\}_{k=1}^{M}$ is a set of edges connecting the vertices.

Each edge $e$ can be identified by the couple of vertices that it connects ($e = (v_i, v_j)$).

The edges simply stand for some type of binary relation between pairs of nodes: in particular, they are not endowed with any geometry.

Only undirected graphs are considered here. In some cases it will be necessary to assign a direction to the edges, but the choice will be arbitrary.
Some matrices associated with combinatorial graphs

- **Adjacency matrix** $A$: symmetric $N \times N$ vertex-to-vertex Boolean structure, $A_{ij} = 1$ iff edge $(v_i, v_j)$ exists.

- **Incidence matrix** $E$: rectangular $N \times M$ vertex-to-edge matrix. Each column corresponds to an edge $e = (v_i, v_j)$ and has only two non zero entries, 1 and $-1$, in position $i$ and $j$ (the sign is arbitrary and will be immaterial for our purposes).

- **Combinatorial Laplacian**: $L_\Gamma = EE^T = D_\Gamma - A$, where $D_\Gamma$ is the diagonal matrix of degrees.
  - $E$ can be interpreted as a discrete divergence and $E^T$ as a discrete gradient.
  - If $\Gamma$ is connected, $\text{Ker}(E^T)$ is the 1D subspace of $\mathbb{R}^N$ spanned by $e = (1, 1, \ldots, 1)^T$.

- $L_\Gamma$ is a **positive semidefinite** singular matrix with $\text{Ker}(L_\Gamma) = \text{span}\{e\}$
Example: graphene sheet
Example: graphene sheet (cont.)
Example: graphene sheet (cont.)
Beyond simple graphs

While combinatorial graphs (including weighted and directed ones) have long been found extremely useful in countless applications, they are too simple for modeling certain types of phenomena on networks.

In many cases, the interaction between pairs of nodes may be more complex than just a 0-1 relation.
For instance, in physics and engineering applications the edges may represent actual physical links between vertices, and these links will typically be endowed with a notion of length.

Hence, communication between nodes may require some time rather than being instantaneous.

This simple observation is formalized in the notion of metric graph.
A graph \( \Gamma \) is a **metric graph** if to each edge \( e \) is assigned a measure (normally the Lebesgue one) and, consequently, a length \( l_e \in (0, \infty) \).

Thus, each edge can be assimilated to a finite interval on the real line \((0, l_e) \subset \mathbb{R}\), with the **natural coordinate** \( s = s_e \).

Note that we need to assign a direction to an edge in order to assign a coordinate to each point on \( e \). For this we can take the (arbitrarily chosen) direction used in the definition of the incidence matrix of \( \Gamma \).

In technical terms, a metric graph is a **topological manifold** (1D simplicial complex) having singularities at the vertices, i.e. it is **not** a differentiable manifold (globally).
The points of a metric graph $\Gamma$ are the vertices, plus all the points on the edges.

The Lebesgue measure is well-defined on all of $\Gamma$ for finite graphs (the only ones considered here). Thus, $\Gamma$ is endowed with a global metric.

The distance between two points (not necessarily vertices) in $\Gamma$ is the length of the geodetic (shortest path) between them.

Note that $\Gamma$ is not necessarily embedded in a Euclidean space $\mathbb{R}^n$. 
The edges may also have physical properties, such as conductivity, diffusivity, permeability etc., that are not well represented by a single scalar quantity, as in a weighted graph. Some of these quantities could even change with time.

In other words, interactions between nodes may be governed by laws, which could be described in terms of differential equations.

One can easily imagine similar situations also for other types of networks, including social or financial networks.

Formalization of this notion leads to the concept of quantum graph.
Hilbert spaces on $\Gamma$

To define a quantum graph, we need first to introduce certain Hilbert spaces of functions defined on a metric graph $\Gamma$.

Let $L^2(e) := \{ f : e \to \mathbb{R} \mid \int_e |f|^2 \, ds < \infty \}$.

**Definition:** The space of square-integrable functions on $\Gamma$ is defined as

$$L^2(\Gamma) := \bigoplus_{e \in \mathcal{E}} L^2(e).$$

In other terms,

$$f \in L^2(\Gamma) \quad \text{iff} \quad \|f\|_{L^2(\Gamma)}^2 = \sum_{e \in \mathcal{E}} \|f\|_{L^2(e)}^2 < \infty.$$
Next, consider the Sobolev space

\[ H^1(e) = \left\{ f \in L^2(e) \mid \int_e |f'(s)|^2 \, ds < \infty \right\}. \]

**Definition:** The Sobolev space \( H^1 \) on \( \Gamma \) is defined as

\[ H^1(\Gamma) = \left( \bigoplus_{e \in \mathcal{E}} H^1(e) \right) \cap C^0(\Gamma) \]

where \( C^0(\Gamma) \) is the space of continuous functions on \( \Gamma \).

In other terms,

\[ f \in H^1(\Gamma) \iff f \text{ is continuous and } \|f\|^2_{H^1(\Gamma)} = \sum_{e \in \mathcal{E}} \|f\|^2_{H^1(e)} < \infty. \]
Quantum graphs (def.)

Let $\mathcal{H}$ be a linear differential operator defined on suitable subspace $D(\mathcal{H}) \subset L^2(\Gamma)$. We will call $\mathcal{H}$ a Hamiltonian on $\Gamma$.

**Definition:** A quantum graph is a metric graph $\Gamma$ together with a Hamiltonian $\mathcal{H}$ and boundary (vertex) conditions that ensure that $\mathcal{H}$ is self-adjoint.

Hence, a quantum graph is a triple $(\Gamma, \mathcal{H}, \text{vertex conditions})$.

In some situations the definition may be altered to allow more general (non-self-adjoint, pseudo-) differential operators.

**Remark:** Although the concept goes back at least to G. Lumer (1980), the name “quantum graph" was introduced by T. Kottos and U. Smilansky (Phys. Rev. Lett. 79 (1997), pp. 4794–4797) and has since become universally adopted.
Hamiltonians

We will focus primarily on the simplest example of a Hamiltonian: the (negative) second derivative operator

\[ u \rightarrow \mathcal{H}u = -\frac{d^2u}{ds^2}. \]

We will also consider Schrödinger operators

\[ u \rightarrow \mathcal{H}u = -\frac{d^2u}{ds^2} + V(s)u, \]

where \( V \) is a potential, usually required to be bounded from below. Here we assume \( u \in H^2(e), \forall e \in \mathcal{E}. \)

More complicated operators can also be considered, for example the magnetic Schrödinger operator

\[ u \rightarrow \mathcal{H}u = \left( \frac{1}{i} \frac{d}{ds} - A(s) \right)^2 u + V(s)u \]

as well as higher order operators, Dirac operators, pseudo-differential operators, etc.
1 Motivation
2 Basic definitions
3 Boundary conditions at the vertices
4 Discretization
5 Numerical experiments
6 Time-dependent problems
7 Conclusions and open problems
In this talk we only consider the so-called Neumann-Kirchhoff conditions, a special case of $\delta$-type conditions:

$$\left\{ \begin{array}{l}
  f(s) \text{ is continuous on } \Gamma \\
  \forall v \in \Gamma \quad \sum_{e \in \mathcal{E}_v} \frac{df}{ds_e}(v) = \alpha_v f(v)
\end{array} \right.$$ 

$\mathcal{E}_v$ is the subset of the edges having $v$ as a boundary point.

The $\alpha_v$'s are fixed real numbers.
The Hamiltonian is associated to the following quadratic form on \( H^1(\Gamma) \):

\[
\mathfrak{h}[f, f] = \sum_{e \in \mathcal{E}} \int_{e} \left| f'(s) \right|^2 ds + \sum_{v \in \mathcal{V}} \alpha_v |f(v)|^2.
\]

The case \( \alpha_v \equiv 0 \) corresponds to the **Neumann-Kirchhoff** conditions:

\[
\begin{align*}
\text{\( f(s) \) is continuous on \( \Gamma \)} & \\
\forall v \in \Gamma & \quad \sum_{e \in \mathcal{E}_v} \frac{df}{ds_e}(v) = 0
\end{align*}
\]

and the corresponding quadratic form reduces to

\[
\mathfrak{h}[f, f] = \sum_{e \in \mathcal{E}} \int_{e} \left| f'(s) \right|^2 ds.
\]
The Neumann-Kirchhoff conditions are also called the standard vertex conditions. They are the natural boundary conditions satisfied by the Schrödinger operator.

The first condition expresses continuity, while the second can be interpreted as conservation of current.
Examples of boundary conditions (cont.)

We observe that, on the other hand, the Dirichlet or Neumann boundary conditions at the vertices are examples of decoupling conditions and are of little interest in this context, except for vertices of degree one.

For example, if we impose the vertex Dirichlet condition $f(v) = 0$ at each vertex, the Hamiltonian is just the direct sum of the operators on each edge $e$ with Dirichlet conditions on the end; hence, the quantum graph decouples into a set of independent intervals, and the topology of the graph becomes irrelevant.
Finite element discretization

Self-adjoint elliptic equations can be formulated as variational problems for an energy functional.

Given a function \( g \in L^2(\Gamma) \), the minimum problem is

\[
\min_{u \in H^1(\Gamma)} J(u),
\]

where

\[
J(u) = \frac{1}{2} \sum_{e \in \mathcal{E}} \int_e \left\{ u'(s)^2 + V(s)u(s)^2 \right\} \, ds - \sum_{e \in \mathcal{E}} \int_e g(s)u(s) \, ds.
\]

We discretize the problem using 1D linear finite elements on each edge and use a domain decomposition approach:

- first we eliminate the unknowns associated with points inside the edges,
- then we use the Neumann-Kirchhoff conditions and the values of the derivative at the vertices to form and solve a reduced-size linear system (Schur complement) for the values of the solution at the vertices.
On each edge of the quantum graph it is possible to use the classical 1D finite-element method. Let $e$ be a generic edge identified by two vertices, which we denote by $v_a$ and $v_b$.

The coordinate $s$ will parameterize the edge such that for $s = 0$ we have the vertex $v_a$ and for $s = \ell_e$ we have the vertex $v_b$.

The first step is to subdivide the edge in $n^e$ intervals of length $h^e$. The points

$$\{ s^e_j \}_{j=1}^{n-1} \cup \{ v_a \} \cup \{ v_b \}$$

form a chain linking $v_a$ to $v_b$ lying on $e$. 
Denoting by \( \{ \psi_j^e \}_{j=0}^{n+1} \) the standard hat basis functions, we have

\[
\begin{aligned}
\psi_0^e(s) &= \begin{cases} 
1 - \frac{s}{h} & \text{if } 0 \leq s \leq h^e \\
0 & \text{otherwise}
\end{cases} \\
\psi_j^e(s) &= \begin{cases} 
1 - \frac{|s_j - s|}{h} & \text{if } s_{j-1} \leq s \leq s_{j+1} \\
0 & \text{otherwise}
\end{cases} \\
\psi_{n+1}^e(s) &= \begin{cases} 
1 - \frac{\ell^e - s}{h} & \text{if } \ell^e - h^e \leq s \leq \ell^e \\
0 & \text{otherwise}
\end{cases}
\end{aligned}
\] (1)

The functions \( \psi_j^e \) are a basis for the finite-dimensional space

\[
V_h^e = \left\{ u_h \in H^1(e) : u_h|_{[s_j^e, s_{j+1}^e]} \in P_1, \quad j = 0, \ldots, n + 1 \right\},
\]

where \( P_1 \) is the space of linear functions.
In practice, we subdivide each edge, forming a chain made of nodes of degree 2, and we build the usual hat functions extending them to the vertices:
Globally, we construct the finite element space

\[ V_h(\Gamma) = \bigoplus_{e \in \mathcal{E}} V^e_h, \]

which is a finite-dimensional subspace of \( H^1(\Gamma) \).

The continuity on \( \Gamma \) of the functions in \( V_h \) follows by construction: at each vertex \( v \) we have \( d_v \) (degree of the vertex \( v \)) linear functions that take the value 1 on \( v \), each one belonging to an independent \( V^e_h \) with \( e \in \mathcal{E}_v \).

Any function \( u_h \in V_h(\Gamma) \) is then a linear combination of the \( \psi^e_j \):

\[ u_h(s) = \sum_{e \in \mathcal{E}} \sum_{j=0}^{n+1} \alpha^e_j \psi^e_j(s). \]
The quadratic form $\mathcal{h}$ of the Hamiltonian operator can be tested on all the $\psi$'s and we have the following finite dimensional (discrete) bilinear form:

$$
\mathcal{h}_h[u_h, \psi^e_k] = \sum_{e \in \mathcal{E}} \sum_{j=0}^{n+1} \alpha^e_j \left\{ \int_e \frac{d\psi^e_j}{ds} \frac{d\psi^e_k}{ds} ds + \int_e V(s)\psi^e_j \psi^e_k ds \right\} .
$$

In both $\mathcal{h}$ and $\mathcal{h}_h$ the Neumann-Kirchhoff conditions at each vertex are the natural conditions and they are automatically satisfied.
The nodes on the edges will describe a chain path between two vertices.

We can then think of introducing a new (combinatorial) graph in which the nodal discretization points become additional vertices and the edges are obtained by subdividing the edges of the original (metric) graph. We call this the extended graph associated with $\Gamma$ and denote it by $\mathcal{G}$.

Assuming for simplicity that all edges $e \in \mathcal{E}$ have equal length and that the same number $n - 1$ of internal nodes are used for each edge, the new graph $\mathcal{G}$ will have $(n - 1) \times M + N$ vertices and $n \times M$ edges, where $N$ is the number of vertices and $M$ the number of edges in $\Gamma$.

The extended graph can be huge, but it has a lot of structure.
It is natural to order the vertices according to the original order of the edges so that the new ("subdomain") vertices on the edges are numbered contiguously, and the vertices of the original graph ("separators") are numbered last.

The resulting Gramian matrix \( \mathbf{H} = (h_{jk}[\psi_j^e, \psi_k^e]) \) is of the form

\[
\mathbf{H} = \begin{bmatrix}
\mathbf{H}_{11} & \mathbf{H}_{12} \\
\mathbf{H}_{12}^T & \mathbf{H}_{22}
\end{bmatrix}
\]

where \( \mathbf{H}_{11} \) is a block diagonal symmetric and positive definite matrix where each diagonal block is of size \( n - 1 \) and tridiagonal, and \( \mathbf{H}_{22} \) is a diagonal matrix with positive diagonal entries.

**Important**: We are assuming that the potential \( V(s) \) is positive.
A simple example

**Figure:** Example of a simple planar metric graph and of its incidence matrix.
A simple example

**Figure:** Example of the extension of the graph when a 4 nodes chain is added internally to each edge (left) and its incidence matrix (right).
Figure: Pattern of the discrete Hamiltonian $H$ where the red bullets correspond to the original vertices and the blue ones to the internal nodes on each edge.
Extended graph (cont.)

In the special case $V = 0$ (that is, $\mathcal{H} = -\frac{d^2}{ds^2}$), we obtain the discrete (negative) Laplacian (stiffness matrix) $L$ on $\mathcal{G}$.

When the same number of (equidistant) discretization points is used on each edge of $\Gamma$, $L$ coincides (up to the factor $h^{-1}$) with the combinatorial graph Laplacian $L_G$.

Both the stiffness matrix $L$ and the mass matrix $M = (\langle \psi^e_j, \psi^e_k \rangle)$ have a block structure matching that of $H$. For example, if $V(s) = k$ (constant) then $H = L + kM$.

Minimization of the discrete quadratic form

$$J_h(u_h) := \mathcal{H}_h[u_h, u_h] - 2\langle g_h, u_h \rangle, \quad u_h \in V_h(\Gamma)$$

is equivalent to solving the extended linear system $H u_h = g_h$, of order $(n - 1)M + N$. 


The extended linear system can be solved efficiently by block LU factorization, by first eliminating the interior edge nodes (this requires solving, in parallel if one wishes, a set of $M$ independent tridiagonal systems of order $n - 1$), and then solving the $N \times N$ Schur complement system

$$S u_h^v = g_h^v - H_{12}^T H_{11}^{-1} f_h^e \equiv c_h$$

for the unknowns associated with the vertices of $\Gamma$. 
Solution of the extended linear system

The block LU factorization of $H$ is given by

$$H = \begin{bmatrix} H_{11} & H_{12} \\ H_{12}^T & H_{22} \end{bmatrix} = \begin{bmatrix} H_{11} & O \\ H_{12}^T & S \end{bmatrix} \begin{bmatrix} I & H_{11}^{-1}H_{12} \\ O & I \end{bmatrix}.$$

Simple, yet crucial observation: the Schur complement

$$S = H_{22} - H_{12}^TH_{11}^{-1}H_{12}$$

is a sparse matrix.
Back to the simple example

Figure: The pattern of the Schur complement $S$. 
Theorem: The nonzero pattern of the Schur complement

\[ S = H_{22} - H_{12}^T H_{11}^{-1} H_{12} \]

coincides with that of \( L_\Gamma \), the (combinatorial) graph Laplacian of the (combinatorial) graph \( \Gamma \).

In the special case \( V = 0 \), we actually have \( S = L_\Gamma \).
Solution of the extended linear system (cont.)

Note that $S$ is SPD, unless $V = 0$ (in which case $S$ is only positive semidefinite).

For $\Gamma$ not too large, we can solve the Schur complement system by sparse Cholesky factorization with an appropriate reordering.

However, for large and complex graphs (for example, scale-free graphs), Cholesky tends to generate enormous amounts of fill-in, regardless of the ordering used.

Hence, we need to solve the Schur complement system by iterative methods, like the preconditioned conjugate gradient (PCG) algorithm.
Preconditioning of matrices arising from complex graphs is an active area of research.

Some of the techniques that work well for other types of problems (like Incomplete Cholesky Factorization) are useless here.

Here we consider two simple preconditioners:

- **diagonal scaling** with \( D = \text{diag}(S) \)
- a first degree **polynomial preconditioner**:

\[
P^{-1} = D^{-1} + D^{-1}(D - S)D^{-1} \approx S^{-1}.
\]

**Note**: For the very sparse matrices considered here, each iteration of PCG with polynomial preconditioning costs about the same as 1.5 iterations with diagonal preconditioning.
Outline

1. Motivation
2. Basic definitions
3. Boundary conditions at the vertices
4. Discretization
5. Numerical experiments
6. Time-dependent problems
7. Conclusions and open problems
Numerical experiments

We present first results for a simple steady-state (equilibrium) problem

$$-\frac{d^2 u}{ds^2} + V u = g \quad \text{on} \quad \Gamma$$

with Neumann-Kirchhoff conditions at the vertices, for three different choices of $\Gamma$:

- yeast, the PPI network of beer yeast ($N = 2224, M = 6609$)
- drugs, a social network of drug addicts ($N = 616, M = 2012$)
- pref2000, a synthetic scale-free graph constructed using the preferential attachment scheme ($N = 2000, M = 3974$)

In each case we assume that all edges have unit length and we use $n = 20$ interior discretization points per edge ($h = \frac{1}{21}$).

For the potential we use $V(s) = k(s - \frac{1}{2})^2$ and $V(s) = k \text{ (const.)}$ for $k = 0.1, 1, 10$. 
PPI network of *Saccharomyces cerevisiae* (beer yeast)
Social network of injecting drug users in Colorado Springs

Figure courtesy of Ernesto Estrada.
Scale-free Barabási–Albert graph (pref)
Numerical experiments (cont.)

The sizes of the extended system $H \mathbf{u}_h = g_h$ and of the reduced system $S \mathbf{u}^v_h = c_h$ are, respectively:

- $n = 134,404, N = 2224$ for yeast;
- $n = 40,856, N = 616$ for drugs;
- $n = 81,480$ and $N = 2000$ for pref2000.

The Schur complement can be formed efficiently since it is very sparse and we know the location of the nonzero entries in advance.

Since the original graphs are very small, the Schur complement system is best solved by sparse Cholesky factorization, but we also experiment with PCG. Without preconditioning, convergence can be slow.

For each problem we also need to solve $M$ uncoupled tridiagonal systems of order 20.
Numerical experiments (cont.)

\[ V(s) = k(s - \frac{1}{2})^2 \]

<table>
<thead>
<tr>
<th>Problem</th>
<th>( k = 0.1 )</th>
<th>( k = 1 )</th>
<th>( k = 10 )</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>( \text{lt} )</td>
<td>( \text{rel. error} )</td>
<td>( \text{lt} )</td>
</tr>
<tr>
<td>yeast</td>
<td>13</td>
<td>3.3e-08</td>
<td>10</td>
</tr>
<tr>
<td>drugs</td>
<td>10</td>
<td>1.4e-08</td>
<td>8</td>
</tr>
<tr>
<td>pref2000</td>
<td>9</td>
<td>2.9e-09</td>
<td>8</td>
</tr>
</tbody>
</table>

Results of running \texttt{pcg} \((TOL = \sqrt{\text{eps}})\) on Schur complement system, diagonal preconditioner.

The “exact” solution is the one returned by backslash.
Numerical experiments (cont.)

\[ V(s) = k(s - \frac{1}{2})^2 \]

<table>
<thead>
<tr>
<th>Problem</th>
<th>( k = 0.1 )</th>
<th>( k = 1 )</th>
<th>( k = 10 )</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>It</td>
<td>rel. error</td>
<td>It</td>
</tr>
<tr>
<td>yeast</td>
<td>7</td>
<td>5.9e-08</td>
<td>6</td>
</tr>
<tr>
<td>drugs</td>
<td>6</td>
<td>3.9e-08</td>
<td>5</td>
</tr>
<tr>
<td>pref2000</td>
<td>5</td>
<td>4.5e-09</td>
<td>5</td>
</tr>
</tbody>
</table>

Results of running \( \text{pcg} \) \((TOL = \sqrt{\text{eps}})\) on Schur complement system, polynomial preconditioner.
Numerical experiments (cont.)

$V(s) = k$

<table>
<thead>
<tr>
<th>Problem</th>
<th>$k = 0.1$</th>
<th>$k = 1$</th>
<th>$k = 10$</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$\text{lt}$</td>
<td>rel. error</td>
<td>$\text{lt}$</td>
</tr>
<tr>
<td>yeast</td>
<td>36</td>
<td>7.7e-09</td>
<td>14</td>
</tr>
<tr>
<td>drugs</td>
<td>42</td>
<td>6.9e-09</td>
<td>14</td>
</tr>
<tr>
<td>pref2000</td>
<td>23</td>
<td>3.5e-09</td>
<td>12</td>
</tr>
</tbody>
</table>

Results of running pcg ($TOL = \sqrt{eps}$) on Schur complement system, diagonal preconditioner.
Numerical experiments (cont.)

<table>
<thead>
<tr>
<th>Problem</th>
<th>( V(s) = k )</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>( k = 0.1 )</td>
</tr>
<tr>
<td></td>
<td>lt</td>
</tr>
<tr>
<td>yeast</td>
<td>20</td>
</tr>
<tr>
<td>drugs</td>
<td>24</td>
</tr>
<tr>
<td>pref2000</td>
<td>13</td>
</tr>
</tbody>
</table>

Results of running \( \text{pcg} (TOL = \sqrt{eps}) \) on Schur complement system, polynomial preconditioner.
Matrix $S^{-1}$ for pref graph, $k = 0.1$. 
Matrix $S^{-1}$ for pref graph, $k = 1$. 
Matrix $S^{-1}$ for \textit{pref} graph, $k = 10$. 
Numerical experiments (cont.)

<table>
<thead>
<tr>
<th>$N$</th>
<th>No prec.</th>
<th>Diagonal</th>
<th>Polynomial</th>
</tr>
</thead>
<tbody>
<tr>
<td>2000</td>
<td>83</td>
<td>23</td>
<td>13</td>
</tr>
<tr>
<td>5000</td>
<td>108</td>
<td>23</td>
<td>13</td>
</tr>
<tr>
<td>10000</td>
<td>125</td>
<td>23</td>
<td>13</td>
</tr>
</tbody>
</table>

PCG iteration counts for pref graph, $V(s) = k = 0.1$, increasing $N$.

**Note:** Here $h$ is constant, the size $N$ of the graph $\Gamma$ is increasing. The size of the extended graph $\mathcal{G}$ is $n = 81,480$, $n = 204,360$, and $n = 409,300$, respectively.
Numerical experiments (cont.)

<table>
<thead>
<tr>
<th>$h^{-1}$</th>
<th>No prec.</th>
<th>Diagonal</th>
<th>Polynomial</th>
</tr>
</thead>
<tbody>
<tr>
<td>21</td>
<td>83</td>
<td>23</td>
<td>13</td>
</tr>
<tr>
<td>41</td>
<td>83</td>
<td>23</td>
<td>13</td>
</tr>
<tr>
<td>81</td>
<td>82</td>
<td>22</td>
<td>12</td>
</tr>
<tr>
<td>101</td>
<td>83</td>
<td>21</td>
<td>12</td>
</tr>
</tbody>
</table>

PCG iteration counts for preff graph, $V(s) = k = 0.1$, $N = 2000$.

**Note:** Here $h$ is decreasing, the size $N$ of $\Gamma$ is fixed. The size of the extended graph $G$ increases from 81,480 to 399,400 vertices.

With diagonal or polynomial preconditioning the solution algorithm is scalable with respect to both $N$ and $h$ for these graphs.
1 Motivation
2 Basic definitions
3 Boundary conditions at the vertices
4 Discretization
5 Numerical experiments
6 Time-dependent problems
7 Conclusions and open problems
The parabolic case

Among our goals is the analysis of diffusion phenomena on metric graphs. In this case, we assume that the functions we use also depend on a second variable \( t \) representing time, i.e.,

\[
  u(t, s) : [0, T] \times \Gamma \longrightarrow \mathbb{R}.
\]

A typical problem would be: given \( u_0 \in H^1(\Gamma) \) and a \( f \in L^2([0, T], L^2(\Gamma)) \) find \( u \in L^2([0, T], H^1(\Gamma)) \cap C^0([0, T]; H^1(\Gamma)) \) such that

\[
\begin{cases}
  \frac{\partial u}{\partial t} - \frac{\partial^2 u}{\partial s^2} + mu = f \quad &\text{on } \Gamma \\
  u(0, s) = u_0,
\end{cases}
\]

where \( m \geq 0 \).

Similarly, we can define the wave equation and the Schrödinger equation on \( \Gamma \).
Space discretization using finite elements leads to the semi-discrete system

\[ M \dot{u}_h = H u_h + f_h, \quad u_h(0) = u_{h,0}, \]

where \( u_h = u_h(t) \) is a vector function on the extended graph \( G \), and the mass matrix \( M \) and Hamiltonian \( H \) are as before.

A variety of methods are available for solving this linear system of ODEs: backward Euler, Crank-Nicolson, exponential integrators based on Krylov subspace methods, etc.

Note that for large graphs and/or small \( h \), this can be a huge system.

We have obtained some preliminary results using Stefan Güttel’s code \texttt{funm_kryl} for evaluating the action of the matrix exponential on a vector.
Solution of diffusion problem on $\Gamma$ for different times.
Solution of diffusion problem on $\Gamma$ for different times.
Outline

1 Motivation
2 Basic definitions
3 Boundary conditions at the vertices
4 Discretization
5 Numerical experiments
6 Time-dependent problems
7 Conclusions and open problems
Summary

- Quantum graphs bring together disparate areas: physics, graph theory, PDEs, spectral theory, complex networks, finite elements, numerical linear algebra...
- On the surface just a huge set of “trivial" 1D problems, but the complexity of the underlying graph and the Neumann-Kirchhoff coupling conditions make life interesting!
- Linear systems are huge but highly structured with much potential for order reduction and parallelism.
- Challenges ahead include:
  - Analyze PCG convergence
  - Eigenvalue problems
  - Hyperbolic problems (shocks)
  - Schrödinger, Dirac equations (important for graphene)
  - Non-self-adjoint and non-linear problems, non-local operators, etc.
  - Applications to real world problems
- There is a lot of work to do in this area!