Weighted Residual Techniques for BVPs

- With finite differences, we approximate the equation, e.g.,
  \[- \frac{d^2 \tilde{u}}{dx^2} = f(x) \rightarrow - \frac{\delta^2 u_j}{\delta x^2} = f_j, \quad + \text{BCs.}\]

- With WRTs, we approximate the solution, e.g.,
  \[u(x) := \sum_{j=1}^{n} \hat{u}_j \phi_j(x), \quad X_N^0 := \text{span}\{\phi_1, \ldots, \phi_n\},\]
  where the \(\phi_j\)s satisfy homogeneous BCs (say).

- We try to make the error, \(e(x) := \tilde{u}(x) - u(x), \quad \text{small.}\)
Collocation

- One way to try to make the error small is to force the residual to be zero at certain collocation points, \( x_i \):

\[
    r(x) := -\frac{d^2u}{dx^2} - f(x) = 0 \text{ at } x_i, \ i = 1, \ldots, n.
\]

\[
    := -\frac{d^2u}{dx^2} + \frac{d^2\tilde{u}}{dx^2}
\]

\[
    := -\frac{d^2e}{dx^2}.
\]

- Clearly, \( r(x) \equiv 0 \) if \( u(x) = \tilde{u}(x) \).

- The residual is computable and is the only available measure of the error.

- Implementation of the collocation scheme is

\[
    \begin{bmatrix}
        \phi''_1(x_1) & \phi''_2(x_1) & \cdots & \phi''_n(x_1) \\
        \phi''_1(x_2) & \phi''_2(x_2) & \cdots & \phi''_n(x_2) \\
        \vdots & \vdots & \ddots & \vdots \\
        \phi''_1(x_n) & \phi''_2(x_n) & \cdots & \phi''_n(x_n)
    \end{bmatrix}
    \begin{bmatrix}
        \hat{u}_1 \\
        \hat{u}_2 \\
        \vdots \\
        \hat{u}_n
    \end{bmatrix}
    =
    \begin{bmatrix}
        f_1 \\
        f_2 \\
        \vdots \\
        f_n
    \end{bmatrix}
\]

- Q: If you were going to implement collocation, what would be a good set of points?
Collocation

- For several reasons, it’s better to make the residual small in a weighted sense, rather than just enforcing \( r(x) = 0 \) at a few isolated points.

- A disadvantage of collocation is that it requires \( \phi_j \in C^1 \), i.e., twice differentiable, which precludes piecewise-linear (FEM) basis functions. :

- Another disadvantage is that it does not guarantee a best-fit approximation.

- Also, it does not yield a symmetric “stiffness” matrix (\( A := -D^2 \)).

- Moreover, Neumann and Robin BCs are not easy to implement (many many papers on this very topic).

- For these reasons, the Weighted Residual Technique is strongly preferred.
Weighted Residual Method

• Here, rather than enforcing \( r(x) = 0 \) pointwise, we seek a solution \( u(x) \in X_0^N \) such that \( r(x) \perp Y^N \) for a suitably chosen \( Y^N \).

• We call \( X_0^N \) the trial space and \( Y_0^N \) the test space.

• Here, we define “\( \perp \)” in the following sense:

\[
\text{Let } (f, g) := \int_{\Omega} f(x) g(x) \, dx.
\]

We say \( f \perp g \) if \( (f, g) = 0 \).

• Q: Why would orthogonality imply a small residual ??
The WRT is essentially a method of undetermined coefficients.

- Consider the 1D Helmholtz equation with $\beta > 0$,
  \[-\frac{d^2 \tilde{u}}{dx^2} + \beta \tilde{u} = f(x), \quad \tilde{u}(0) = \tilde{u}(1) = 0.\]

- Seek an approximate solution $u$ in a finite-dimensional trial space $X_0^N$,
  \[u \in X_0^N := \text{span}\{\phi_1(x), \phi_1(x), \ldots, \phi_n(x)\}; \quad \phi_j(0) = \phi_j(1) = 0.\]
(We use the subscript 0 on $X_0^N$ to indicate that functions in this space satisfy the homogeneous Dirichlet boundary conditions.)
• The trial solution is a linear combination of the \textit{basis functions} \( \phi_j(x) \) with \textit{basis coefficients} \( \hat{u}_j \),

\[
  u(x) = \sum_{j=1}^{n} \phi_j(x) \hat{u}_j.
\]

• The orthogonality condition is based on the standard \( L^2 \) inner product. Specifically, we require

\[
  0 = \int_0^1 v(x) r(x) \, dx = \int_0^1 v(x) \left( f + \frac{d^2 u}{dx^2} - \beta u \right) \, dx \quad \forall v \in Y_0^N
\]

or,

\[
  \int_0^1 v(x) \left( -\frac{d^2 u}{dx^2} + \beta u \right) \, dx = \int_0^1 v f \, dx \quad \forall v \in Y_0^N.
\]

• Note that if \( Y_0^N=\text{span}\{\psi_i(x)\}, \ i = 1, \ldots, n \) and \( \psi_i(x) = \delta(x - x_i) \), the Dirac delta function, then we recover \textbf{collocation}.

  – That is, we are enforcing \( r(x_i)=0 \).
**Galerkin Method**

- For the Poisson and Helmholtz equations, the optimal choice is $Y_0^N = X_0^N$, which is the **Galerkin method**.

- It appears that $u$ must be twice differentiable.
  We can avoid this requirement through integration by parts.

- Let $I$ denote the left-hand side of the preceding equation.

\[
I = \int_0^1 \left( -v(x) \frac{d^2u}{dx^2} + \beta vu \right) dx
\]

\[
= \int_0^1 \left( \frac{dv}{dx} \frac{du}{dx} + \beta vu \right) dx - \left. vu \right|_0^1
\]

\[
= \int_0^1 \left( \frac{dv}{dx} \frac{du}{dx} + \beta vu \right) dx.
\]

- The boundary terms vanish because $v = 0$ at $x = 0$ and 1.

- We see that the number of derivatives on the trial ($u$) and test ($v$) functions is now the same.

- They thus have the same (low) continuity requirements, which is feasible because they are in the same space ($Y_0^N \equiv X_0^N$).
• We denote the integral $\mathcal{I}$ as the energy (or “$a$”) inner-product,
\[ a(v, u) := \int_0^1 \left( \frac{dv}{dx} \frac{du}{dx} + \beta vu \right) dx. \]

• $a(\cdot, \cdot)$ is symmetric, $a(v, u) = a(u, v)$, and positive definite:
\[ a(u, u) > 0 \forall u \neq 0. \]

• Our discrete problem can be stated as, \textit{Find $u \in X_0^N$ such that}
\[ a(v, u) = (v, f) \forall v \in X_0^N. \]

• Note that this statement is an identity for $\tilde{u}$ (generally $\tilde{u} \not\in X_0^N$):
\[ a(v, \tilde{u}) \equiv (v, f) \]
which holds for all $v$ for which the integrand is computable.
• So we have, \( \text{Find } u \in X_0^N \text{ such that, for all } v \in X_0^N, \)
\[
    a(v, u) = a(v, \tilde{u})
\]
\[
    a(v, \tilde{u}) - a(v, u) = 0
\]
\[
    a(v, \tilde{u} - u) = 0.
\]

• Which implies that \( e := \tilde{u} - u \) is \( a \)-orthogonal to \( X_0^N \): \( e(x) \perp_{a} X_0^N \):

\begin{itemize}
\item Therefore, \( u(x) \) is the closest function in \( X_0^N \) to \( \tilde{u}(x) \) in the \( \| \cdot \|_a \) norm.

  – For any function \( w \) satisfying the homogeneous Dirichlet conditions, \( w(0) = w(1) = 0 \), we define the “\( a \)-norm”

    \[
    \| w \|_a := \sqrt{a(w, w)}.
    \]
\end{itemize}
Implementation

• We start with evaluating the terms in our weighted residual statement:

\[ a(v, u) = (v, f), \quad \forall v \in X_0^N := \text{span}\{\phi_1, \ldots, \phi_n\}. \]

• For simplicity, let’s consider the case \( \beta = 0 \).

• Set \( v = \phi_i, \ i = 1, \ldots, n. \)

• Equation \( i : \ a(\phi_i, u) = (\phi_i, f) : \)

\[
\int \frac{d\phi_i}{dx} \frac{du}{dx} \, dx = \int \phi_i \, f \, dx
\]

\[
\int \frac{d\phi_i}{dx} \left( \sum_{j=1}^{n} \frac{d\phi_j}{dx} \, u_j \right) \, dx = \int \phi_i \, f \, dx = b_i
\]

\[
\sum_{j=1}^{n} \left( \frac{\int \frac{d\phi_i}{dx} \frac{d\phi_j}{dx} \, dx}{a_{ij}} \right) u_j = b_i
\]

\[
\sum_{j=1}^{n} a_{ij} \, u_j = b_i
\]

\[ Au = b. \]

• Our task is to evaluate the integrals

\[ a_{ij} := \int \frac{d\phi_i}{dx} \frac{d\phi_j}{dx} \, dx, \quad b_i := \int \phi_i \, f \, dx, \]

and to solve the linear system \( Au = b. \)
Integration

- $\int f g \, dx$: Use quadrature – or not.

- For spectral/ spectral element method – use GLL quadrature.

- For finite element method – integrate exactly (if possible).
  
  - Meaning: integrate function pairs $(f, g)$ exactly for $f, g \in X_0^N$.

Quadrature

- For now, we consider the quadrature-based approach.

$$I = \int_a^b f(\xi_k) g(\xi_k) w_k$$

$$x_k = a + \frac{L_x}{2} (\xi_k + 1), \quad w_k = \frac{L_x}{2} \rho_k. \quad L_x := (b - a),$$

- $\xi_k$ and $\rho_k$ are the Gauss-Lobatto-Legendre points and weights on $[-1, 1]$. 
Example

• Stiffness matrix:

\[ a_{ij} := \int \frac{d\phi_i}{dx} \frac{d\phi_j}{dx} dx, \]

• **Note:** GLL quadrature *exact* for \( f \cdot g \in \mathbb{P}_{2N-1} \).

• Therefore, with Lagrange-polynomial bases, \( \phi_i(x) = l_i(x) \in \mathbb{P}_N, l_i(x_j) = \delta_{ij}, \)

\[ \frac{d\phi_i}{dx} \in \mathbb{P}_{N-1}, \quad \frac{d\phi_i d\phi_j}{dx dx} \in \mathbb{P}_{2N-2} \subset \mathbb{P}_{2N-1}. \]

• Stiffness matrix is exact:

\[ a_{ij} = \int \frac{d\phi_i}{dx} \frac{d\phi_j}{dx} dx, \quad \equiv \sum_{k=0}^{N} \frac{d\phi_i}{dx} \bigg|_{x_k} \frac{d\phi_j}{dx} \bigg|_{x_k} w_k \]

\[ = \sum_{k=0}^{N} D_{ki} w_k D_{kj}. \]

• Let \( B_{ij} := w_i \delta_{ij} \) (diagonal mass matrix). Then,

\[ a_{ij} = \sum_{l=0}^{N} \sum_{k=0}^{N} D_{ki} B_{kl} D_{lj} \]

\[ = \sum_{l=0}^{N} \sum_{k=0}^{N} D_{ik} B_{kl} D_{lj} \]

\[ A = D^T BD. \]

• This is the standard form of the stiffness matrix.

• We will use it *often*, as it is a basic building block.

• Note *carefully* the dimensions of the matrices
  (i.e., the range of indices for all terms in the sums).
A Note on Domain Transformations

- Consider

\[ x \in [a, b] =: \Omega \quad r \in [-1, 1] =: \hat{\Omega} \]

(sometimes use \( r \equiv \xi \))

\[ x(r) = a + \frac{b - a}{2} (r + 1) \quad r(x) = 2 \frac{x - a}{b - a} - 1 \]

\[ = a + \frac{L_x}{2} (r + 1) \quad = 2 \frac{x - a}{L_x} - 1 \]

\[ \frac{dx}{dr} = \frac{L_x}{2} \quad \frac{dr}{dx} = \frac{2}{L_x} \]

- Differentiation with respect to \( x \):

\[ u(x) = \sum_{j=0}^{N} l_j(r) u_j = \sum_{j=0}^{N} l_j(r(x)) u_j \]

\[ \frac{du}{dx} = \sum_{j=0}^{N} \frac{d}{dx} [l_j(r)] u_j = \sum_{j=0}^{N} \frac{dl_j}{dr} \frac{dr}{dx} u_j \]

\[ = \frac{2}{L_x} \sum_{j=0}^{N} \frac{dl_j}{dr} u_j \]

- Evaluating derivative at nodal points, \( x_i \) (or equivalently, \( \xi_i \)):

\[ \left. \frac{du}{dx} \right|_{x_i} = \frac{2}{L_x} \sum_{j=0}^{N} \left( \frac{dl_j}{dx} \right)_{\xi_j} \]

\[ u' = \frac{2}{L_x} \hat{D} u. \]
Transformation from Reference Domain, $\hat{\Omega}$

- Let $\xi_i$ and $\rho_i$ be the standard GLL points and weights on $\hat{\Omega} := [-1, 1]$.
- Our diagonal mass matrix $B$ is a multiple of $\hat{B}$:
  \[
  w_i = \frac{b-a}{2} \rho_i = \frac{L_x}{2} \rho_i \\
  B = \frac{L_x}{2} \hat{B}, \quad \hat{B}_{ij} := \rho_i \delta_{ij},
  \]
  where $\hat{B}$ is the mass matrix on $\hat{\Omega}$.
- Define the derivative matrix:
  \[
  D_{ij} = \left. \frac{2}{b-a} \hat{D}_{ij} \right|_{\xi_j}, \quad \hat{D}_{ij} := \frac{dl_j}{dr} \bigg|_{\xi_j},
  \]
  \[
  l_j(r) \in \mathbb{P}_N, \quad l_j(\xi_j) = \delta_{ij}.
  \]
  where $\hat{D}$ is the derivative matrix on $\hat{\Omega}$
  \[
  \hat{D}_{ij} = \left. \frac{dl_j}{dr} \right|_{\xi_j}, \quad \left\{ \begin{array}{l}
  j = 1, \ldots, N-1 \\
  i = 0, \ldots, N
  \end{array} \right.
  \]
- So, for $i = 1, \ldots, n$ ($n := N - 1$),
  \[
  a_{ij} = \frac{2}{b-a} \hat{a}_{ij} = \frac{2}{L_x} \left[ \hat{D}^T \hat{B} \hat{D} \right]_{ij}
  \]
- That is, our stiffness matrix is:
  \[
  A = \frac{2}{L_x} \hat{D}^T \hat{B} \hat{D}.
  \]
- **NOTE:**
  - $(\xi_i, \rho_i)$ come from the `zwgll.m` utility already used in the HW.
  - $\hat{D}$ comes from one of your early HW assignments.
  - So you have all the tools required to solve a large class of problems.
Extension 1:

- Solve the following BVP with functions $p(x) > 0$ and $q(x) \geq 0$,

$$-\frac{d}{dx}p(x)\frac{du}{dx} + q(x)u = f(x), \quad u(a) = u(b) = 0.$$ 

- WRT: Find $u \in X_0^N$ such that for all $v \in X_0^N$

$$-\int v \left[ \frac{d}{dx}p(x)\frac{du}{dx} + q(x)u \right] \, dx = \int v f \, dx$$

$$-\int v \left[ \frac{d}{dx}p(x)\frac{du}{dx} \right] \, dx + \int vq(x) \, dx = \int v f \, dx$$

$$\int \frac{dv}{dx}p(x)\frac{du}{dx} \, dx - \int v p\frac{du}{dx} \, dx \bigg|_{a}^{b} + \int vq(x) \, dx = \int v f \, dx$$

$$H u = b$$

Note that the boundary term from the integration-by-parts vanishes because $v \in X_0^N$.

- Using quadrature to evaluate all the matrix entries, we find:

$$b_i := \int \phi_i f \, dx \approx \frac{L_x}{2} \hat{B} f \ (i\text{th entry}),$$

$$H = \frac{2}{L_x} \hat{D}^T \left[ \hat{P} \cdot \hat{B} \right] \hat{D} + \frac{L_x}{2} \begin{bmatrix} \hat{B}Q \end{bmatrix} \text{ diagonal} .$$

- Because $p(x) > 0$ and $q(x) \geq 0$, $H$ is SPD.
Example 1: Oscillatory Diffusion Coefficient

• Consider
\[-\frac{d}{dx} e^{\sin k\pi x} \frac{du}{dx} + q(x)u = f(x), \quad u(0) = u(1) = 0.\]

• Manufactured solution: \( u = \sin l\pi x. \)

\[
\begin{align*}
  u' &= l\pi \cos l\pi x \\
  w &= e^{\sin k\pi x} u' \\
  -w' &= e^{\sin k\pi x} \pi^2 \left[ l^2 \sin l\pi x - lk \cos k\pi x \cos l\pi x \right] \\
  &= f(x).
\end{align*}
\]
Solve \(-\frac{d}{dx}|e^{\sin k\pi x}|\frac{du}{dx}\) = \(f(x),\ u(0)=u(1)=0\) with exact solution \(u(x)=\sin l\pi x\).

```matlab
% format compact; format longe; close all; clear all
a=0; b=1; Lx=(b-a); k=3; l=8;
for N=4:80;
    [z,w]=zwgll(N); Bh=diag(w); Dh=deriv_mat(z,z);
    B=(Lx/2)*Bh; D=(2/Lx)*Dh;
    x=a+(Lx/2)*(z+1); % Includes endpoints
    sk=sin(k*pi*x); ck=cos(k*pi*x);
    sl=sin(l*pi*x); cl=cos(l*pi*x);
    e =exp(sk);
    f = (pi*pi)*e.*(l*l*sl-l*k*ck.*cl); b = B*f; % RHS
    P = diag(e); A = D'*(P*B)*D; % Stiffness Matrix
    A = A(2:(end-1),2:(end-1)); b = b(2:(end-1)); % Interior only
    u = A\b; u=[0; u ; 0]; % Solve & add zero bcs
    ut=sl;
    err=max(abs(ut-u)); [N err]
    nn(N-3)=N; en(N-3)=err;
    plot(x,ut,'r--',x,u,'b-','linewidth',2)
    xlabel('x','fontsize',14); ylabel('u','fontsize',14);
    title('Galerkin Solution','fontsize',14); axis([0 1 -1.1 1.1]);
    pause(.1)
end;
figure
semilogy(nn,en,'k-')
```

Figure 1: Solution and error behavior for Galerkin solution of Example 1.
Example 2: Poisson Equation in Cylindrical Coordinates.

- Consider the generalized eigenvalue problem:
  \[-\frac{1}{r} \frac{d}{dr} r \frac{du}{dr} = \lambda u, \quad u'(0) = u(1) = 0.\]

- Galerkin discretization:
  \[A u = \lambda \begin{bmatrix} BQ \\ \text{diagonal} \end{bmatrix} u, \quad Q_{ij} = r_i \delta_{ij},\]
  \[A = D^T PBD, \quad P_{ij} = r_i \delta_{ij},\]

- Neumann condition at \( r = 0 \) (new).
- Dirichlet condition at \( r = 1 \).

- After some manipulation, can show that \( \min \lambda = \tilde{r}_0^2 \), where \( \tilde{r}_0 \) is the first zero of the zeroth-order Bessel function, \( J_0(r) \).
- That is, if \( \tilde{u} = J_0(r) \) is the solution of
  \[\frac{d^2 \tilde{u}}{dr^2} + \frac{1}{r} \frac{d \tilde{u}}{dr} + \tilde{u} = 0.\]

- Then, with the substitution \( \tilde{r} = \tilde{r}_0 r, \ u(r) := \tilde{u}[\tilde{r}(r)] \), one finds,
  \[\frac{d^2 u}{dr^2} + \frac{1}{r} \frac{d u}{dr} + \lambda u = 0, \quad u'(0) = u(1) = 0,\]
  with \( \lambda = \tilde{r}_0^2.\)

- See \texttt{bess.eig.m} example, next slide.
format compact; format long e; close all; clear all

% Solve \[ \frac{d}{dr} \left( r \frac{du}{dr} \right) = \lambda r u, \quad u'(0) = u(1) = 0 \]

a = 0; b = 1; Lx = (b-a);

r0 = 2.40482555769577; %% First zero of J0
zz = 0:.001:1;
ut = besselj(0,r0*zz);
for N=2:18;
    [z,w] = zwgll(N); Bh = diag(w); Dh = deriv_mat(z,z);
    B = (Lx/2)*Bh; D = (2/Lx)*Dh;
    x = a+(Lx/2)*(z+1); % Includes endpoints
    R = diag(x);
    A = D'*R*B*D; %% Stiffness Matrix
    B = R*B; %% Weighted LHS
    A = A(1:(end-1),1:(end-1)); %% Interior: include r=0, exclude r=1
    B = B(1:(end-1),1:(end-1)); %% Interior: include r=0, exclude r=1
    [V,d] = eig(A,B); d = diag(d);
    [d,i] = sort(d); V = V(:,i); u = [V(:,1);0]; if u(1)<0; u = -u; end;
    plot(x,u,'r-',zz,ut,'b--');
    xlabel('x','fontsize',14); ylabel('1st Eigenmode','fontsize',14);
    title('Galerkin Solution Bessel Function','fontsize',14);
    axis square; pause(.1);
end

set(gcf, 'PaperUnits', 'normalized')
set(gcf, 'PaperPosition', [0 0 1 1])
print -dpdf 'bessel_u.pdf'
figure; semilogy(nn,en,'k.-','linewidth',3)
xlabel('N','fontsize',14); ylabel('max pointwise error','fontsize',14);
title('Galerkin Solution Bessel Function','fontsize',14);
axis square
set(gcf, 'PaperUnits', 'normalized')
set(gcf, 'PaperPosition', [0 0 1 1])
print -dpdf 'bessel_e.pdf'
Boundary Conditions Other Than $u(1)=0$.

- One of the attractions of Galerkin methods is the relative ease with which one can implement a variety of boundary conditions.
- Indeed, it is this feature where it distinguishes itself from finite difference and collocation methods, particularly for the high-order case.
- Here, we consider several boundary conditions for the elliptic model problem:

$$- \frac{d}{dx} p(x) \frac{du}{dx} + q(x) u = f(x)$$

- Specifically, we consider:

  Inhomogeneous Dirichlet: $u(a) = u_a, u(b) = u_b$
  Homogeneous Neumann: $u(0) = 0, u'(b) = 0$
  Inhomogeneous Neumann: $u(0) = 0, u'(b) = g_b$
  Robin: $u(0) = 0, u'(b) + \beta u(b) = \gamma$
  Periodic: $u(a) = u(b)$
Inhomogeneous Dirichlet

- Consider
  \[- \frac{d}{dx} p(x) \frac{d\tilde{u}}{dx} + q(x) u = f(x), \quad \tilde{u}(a) = u_a, \quad \tilde{u}(b) = u_b.\]

- Let \( \tilde{u}(x) := \sum_{j=0}^{N} u_j \phi_j(x) = \sum_{j=0}^{N} u_j l_j(r(x)), \quad r = 2\frac{x-a}{b-a} - 1.\)

- Note the range of indices on \( j \) and that \( u_0 = u_a, \quad u_N = u_b.\)

- Since our BCs give us two equations, we only need \((N + 1) - 2 = N - 1\) additional equations.

- For \( i = 1, \ldots, N - 1,\)
  \[\int_a^b d\phi_i \frac{d\tilde{u}}{dx} dx + \int_a^b \phi_i q(x) \tilde{u}(x) dx - \phi_i p(x) \frac{d\tilde{u}}{dx} \bigg|_a^b = \int_a^b \phi_i f(x) dx.\]

- Inserting our expansion for \( \tilde{u} \), coupled with WRT matrices yields
  \[\sum_{j=0}^{N} \left[ A_{ij} + (BQ)_{ij} \right] u_j = w_i f_i =: b_i, \quad i = 1, \ldots, N - 1.\]
  \[- \sum_{j=0}^{N} \tilde{H}_{ij} u_j = b_i.\]
Inhomogeneous Dirichlet (cont’d)

• Or, in compact form, $\bar{H}\bar{u} = \bar{b}$. (Note that $\bar{H}$ is not square!)

\[
\begin{bmatrix}
  h_0 & h_1 & \cdots & h_N
\end{bmatrix}
\begin{bmatrix}
  u_0 \\
  u_1 \\
  \vdots \\
  u_{N-1} \\
  u_N
\end{bmatrix}
= 
\begin{bmatrix}
  b_1 \\
  b_2 \\
  \vdots \\
  b_{N-1}
\end{bmatrix}.
\]

$\bar{H} \in \mathbb{R}^{(N-1) \times (N+1)}$

• As we did in the finite-difference case, we can subtract off the first and last columns:

\[
\begin{bmatrix}
  h_1 & h_2 & \cdots & h_{N-1}
\end{bmatrix}
\begin{bmatrix}
  u_1 \\
  u_2 \\
  \vdots \\
  u_{N-1}
\end{bmatrix}
= 
\begin{bmatrix}
  b_1 \\
  b_2 \\
  \vdots \\
  b_{N-1}
\end{bmatrix} - h_0 u_0 - h_N u_N.
\]

$H \in \mathbb{R}^{n \times n}$

• Here, we define $n = (N - 1)$ such that $H$ is a square, $n \times n$ matrix.

• Note that $H$ is SPD and therefore guaranteed invertible.
Homogeneous Neumann Conditions

- Consider
  \[-\frac{d}{dx}p(x)\frac{d\tilde{u}}{dx} + q(x)u = f(x), \quad \tilde{u}(0) = 0, \quad \tilde{u}'(1) = 0.\]

- Now we allow \( u(x) \) to fluctuate at \( x = 1 \):
  \[ u(x) = \sum_{j=1}^{N} u_j l_j(r), \quad \in X_0^N := \text{span}\{l_1, \ldots, l_N\}. \]

- WRT: Find \( u \in X_0^N \) such that \( \forall v \in X_0^N, \)
  \[ \int_{0}^{1} \frac{dv}{dx}p(x)\frac{du}{dx} - v\frac{du}{dx}\bigg|_{0}^{1} + \int_{0}^{1} v q u dx = \int_{0}^{1} v f dx. \]

- Linear system:
  \[ (A + BQ)u = Bf. \]

- This is the system we solve.

- Notice that the unknowns run from \( j = 1 \) (first interior point on left) to \( j = N \) (last point, on the domain boundary).

- We are thus testing this equation at all interior points \textit{and} at the right domain boundary.
Homogeneous Neumann Conditions, cont’d

• Miraculously, this system looks just like the Dirichlet problem!

• How does it work?

• Consider

\[
\begin{align*}
  a(v, u) &= a(v, \tilde{u}) \\
  a(v, u) := \int_0^1 \frac{dv}{dx} &p(x)\frac{du}{dx} + \int_0^1 v q u \, dx.
\end{align*}
\]

• Subtract rhs from lhs of first equation and integrate by parts

\[
0 = a(v, u) - a(v, \tilde{u}) = a(v, u - \tilde{u})
\]

\[
= -\int_0^1 v p \left. \frac{d^2}{dx^2} (u - \tilde{u}) - v q (u - \tilde{u}) \right. \, dx + v p \left. \frac{d}{dx} (u - \tilde{u}) \right. \bigg|_0^1
\]

\[
= \int_0^1 v \left( -p \frac{du^2}{dx^2} + q u - f \right) \, dx + v p \left. \frac{du}{dx} \right|_{x=1}.
\]

• With quadrature, the Nth-equation reads, \( v = l_N(r) \)

\[
w_N \left[ -p \frac{d^2 u}{dx^2} + q u - f \right] + p \left. \frac{du}{dx} \right|_{x=1} = 0.
\]

• We have \( \rho_N \rightarrow 0 \) as \( N \rightarrow \infty \).

• Thus, as we refine the mesh, the Nth equation drives

\[
\left. \frac{du}{dx} \right|_{x=1} \rightarrow 0.
\]

• This is weak imposition of the Neumann boundary condition.
Inhomogeneous Neumann Conditions

- Here: \( \frac{du}{dx}\big|_b = g_b \).

- WRT:

\[
\int_a^b v \left( -p \frac{du^2}{dx^2} + q u \right) \, dx = \int_a^b v f \, dx \quad \forall \, v \in X_0^N.
\]

- Integrate by parts:

\[
\int_a^b \frac{dv}{dx} p(x) \frac{du}{dx} \, dx - v p \frac{du}{dx}\big|_0^1 + \int_a^b v q u \, dx = \int_a^b v f \, dx.
\]

\[
- v p g_b
\]

- For \( v = l_i(r), i = 1, \ldots, N \), only \( l_N(b) \neq 0 \).

- Full set of equations, taking \( i = 1, \ldots, N \),

\[
\sum_{j=1}^N \left[ \int_a^b \frac{d\phi_i}{dx} p(x) \frac{d\phi_j}{dx} + \phi_i \phi_j q \, dx \right] u_j = \int_a^b \phi_i f \, dx + \phi_i(x_N) p(x_N) g_b. \]

- Matrix form:

\[
Hu = Bf + \begin{pmatrix} 0 \\ 0 \\ \vdots \\ w_N P_N g_b \end{pmatrix},
\]

where \( w_N = \frac{L_x}{2} \rho_N \) is the quadrature weight at \( x_N \).
Robin Conditions

- Consider

\[ u(a) = 0 \]
\[ \frac{du}{dx} + \beta u = \gamma \text{ at } x = b. \]

- WRT:

\[ \int_a^b v \left( -p \frac{du^2}{dx^2} + q u \right) \, dx = \int_a^b v f \, dx \quad \forall v \in X_0^N. \]

- Integrate by parts:

\[ \int_a^b \frac{dv}{dx} p(x) \frac{du}{dx} \, dx + \int_a^b v q u \, dx - \left. v p \frac{du}{dx} \right|_a^b = \int_a^b v f \, dx. \]

\[ (\gamma - \beta u) \]

\[ \int_a^b \frac{dv}{dx} p(x) \frac{du}{dx} \, dx + \int_a^b v q u \, dx + \left. v p \beta u \right|_b = \int_a^b v f \, dx + \left. v p \gamma \right|_b \]

- Matrix form:

\[
Hu + \begin{bmatrix}
0 & & \\
& \ddots & \\
& & \w_{NNpN} \beta
\end{bmatrix}
\begin{bmatrix}
u_1 \\
u_2 \\
\vdots \\
u_N
\end{bmatrix}
= Bf + \begin{bmatrix}
0 \\
0 \\
\vdots \\
\w_{NNpN} \gamma
\end{bmatrix},
\]

- Q: Any constraints on \( \beta \)?
Periodic Conditions

• Here, \( u_0 \equiv u_N, \ v_0 \equiv v_N \).
• Consider the matrix form for \( a(v, u) \) with \( q(x) \equiv 0 \) (just for simplicity),

\[
\begin{align*}
\text{For } v(x) &= \sum_{i=0}^{N} v_i \phi_i(x) \quad u(x) = \sum_{j=0}^{N} u_j \phi_j(x), \\
a(v, u) := & \int_{a}^{b} dv \frac{d}{dx} p(x) \frac{du}{dx} dx \\
&= \int_{a}^{b} \frac{d}{dx} \left( \sum_{i=0}^{N} v_i \phi_i(x) \right) \frac{d}{dx} \left( \sum_{j=0}^{N} u_j \phi_j(x) \right) dx \\
&= \sum_{i=0}^{N} \sum_{j=0}^{N} v_i \left( \int_{a}^{b} \frac{d\phi_i}{dx} \frac{d\phi_j}{dx} dx \right) u_j \\
&= \sum_{i=0}^{N} \sum_{j=0}^{N} v_i A_{ij} u_j \\
&= \bar{v}^T \bar{A} \bar{u}.
\end{align*}
\]
• Now we want use the fact that \( u_0 \equiv u_N \) and \( v_0 \equiv v_N \).
Consider the following matrix-vector multiplication:

\[
\begin{pmatrix}
  u_0 \\
  u_1 \\
  u_2 \\
  u_3 \\
  u_N
\end{pmatrix}
= \begin{pmatrix}
  u_N \\
  u_1 \\
  u_2 \\
  u_3 \\
  u_N
\end{pmatrix}
= \begin{bmatrix}
  0 & 0 & 0 & 1 \\
  1 & 0 & 0 & 0 \\
  0 & 1 & 0 & 0 \\
  0 & 0 & 1 & 0 \\
  0 & 0 & 0 & 1
\end{bmatrix}
\begin{pmatrix}
  u_1 \\
  u_2 \\
  u_3 \\
  u_N
\end{pmatrix}.
\]

=: Q

We see that the action of the rectangular matrix Q is to copy \( u_N \) to the first location in the vector on the left.

Thus, \( \bar{u} := Qu \) produces a vector whose first and last components are the same, exactly as required for the periodic boundary condition case.

Thus, if we define \( \bar{A} \) to be the matrix that includes both endpoints,

\[ \bar{A} := A_{ij}, \quad i, j \in \{0, \ldots, N\}^2, \]

we can define the periodic stiffness matrix

\[ A = Q^T \bar{A} Q. \]

To see this more clearly, let \( \bar{u} := (u_0, u_1, \ldots, u_N)^T = (u_N, u_1, \ldots, u_N)^T \) be the vector of unknown basis coefficients that includes the (periodic) endpoints.

Let \( u := (u_1, \ldots, u_N)^T \) be the vector of unique unknown basis coefficients.

Then \( \bar{u} = Qu \), by the construction of Q.

Similarly, \( \bar{v} = Qv \).

Thus,

\[ a(v, u) = \bar{v}^T \bar{A} \bar{u} = (Qv)^T \bar{A} (Qu) \]

\[ = v^T Q^T \bar{A} Qu \]

\[ = v^T \underbrace{(Q^T \bar{A} Q)}_{=: A} u. \]

The \( N \times N \) matrix A is symmetric semi-definite.

- For example, \( A e = 0 \), where \( e = (1, 1, \ldots, 1)^T \) is the vector of all 1s.