# Outline

1. Boundary Value Problems

2. Numerical Methods for BVPs

# Boundary Value Problems

- Side conditions prescribing solution or derivative values at specified points are required to make solution of ODE unique
- For initial value problem, all side conditions are specified at single point, say $t_0$
- For *boundary value problem* (BVP), side conditions are specified at more than one point
- $k$th order ODE, or equivalent first-order system, requires $k$ side conditions
- For ODEs, side conditions are typically specified at endpoints of interval $[a, b]$, so we have *two-point boundary value problem* with boundary conditions (BC) at $a$ and $b$.

# ODEs: Boundary Value Problems in 1D

Consider linear ODE of the form $\mathcal{L}\tilde{u} = f(x)$, with $\tilde{u}(x)$ satisfying given BCs.

Here, we consider three basic approaches to find $u \approx \tilde{u}$.

- **Finite difference methods (FDM):**
  - Essentially, approximate differential equation at multiple points, $x_i$, $i = 0, \ldots, n$.
  - **Note:** *we will use either $n$ or $n+1$ points according to what makes the most sense in the given context.*

- **Collocation methods:**
  - Approximate the solution by an expansion,

$$u(x) := \sum_{j=0}^{n} u_j \phi_j(x),$$

  - Solve for coefficients $u_j$ such that the ODE is satisfied at a chosen set of collocation points, $x_i$, along with the boundary conditions.
  - That is, the residual, $r(x) := (L\tilde{u} - Lu)$ is forced to be zero at $x_i$, $i = 0, \ldots, n$.

- **Weighted residual technique (WRT):**
  - Approximate the solution by an expansion,

    $$u(x) \ := \ \sum_{j=0}^{n} u_j \phi_j(x),$$

    and solve for coefficients $u_j$ such that the ODE is satisfied in some weighted sense.
  - That is, rather than enforcing $r(x) = 0$ at isolated points, we require $r(x)$ to be orthogonal to a set of weight functions, $\psi_i(x)$:

    $$\int_a^b \psi_i(x)\, r(x)\, dx \ = \ \int_a^b \psi_i(x)\, L(u) - L(\tilde{u})\, dx \ = \ 0, \quad \text{or}$$

    $$\int_a^b \psi_i(x)\, L(u) \ = \ \int_a^b \psi_i(x)\, L(\tilde{u})\, dx$$

    for $i = 0,\ 1,\ \ldots$
  - Note that if $\psi_i(x) = \delta(x - x_i)$ (Dirac delta function), we recover collocation.
  - Most often, the *test-space* and *trial space* are the same: $\psi_i := \phi_i$. (Galerkin case.)
  - Finite element, spectral, spectral element methods are examples of WRTs.

  - WRTs have many advantages over collocation in terms of flexibility of basis functions, application of boundary conditions, etc., and are generally preferred over collocation.

# Finite Difference Method

- *Finite difference method* converts BVP into system of algebraic equations by replacing all derivatives with finite difference approximations
- For example, to solve two-point BVP

  $$u'' = f(t, u, u'), \qquad a < t < b$$

  with BC

  $$u(a) = \alpha, \qquad u(b) = \beta$$

  we introduce mesh points $t_i = a + ih$, $i = 0, 1, \ldots, n+1$, where $h = (b-a)/(n+1)$
- We already have $y_0 = u(a) = \alpha$ and $y_{n+1} = u(b) = \beta$ from BC, and we seek approximate solution value $y_i \approx u(t_i)$ at each interior mesh point $t_i$, $i = 1, \ldots, n$

# Finite Difference Method, continued

- We replace derivatives by finite difference approximations such as

$$u'(t_i) \approx \frac{y_{i+1} - y_{i-1}}{2h}$$

$$u''(t_i) \approx \frac{y_{i+1} - 2y_i + y_{i-1}}{h^2}$$

- This yields system of equations

$$\frac{y_{i+1} - 2y_i + y_{i-1}}{h^2} = f\left(t_i, y_i, \frac{y_{i+1} - y_{i-1}}{2h}\right)$$

  to be solved for unknowns $y_i$, $i = 1, \ldots, n$

- System of equations may be linear or nonlinear, depending on whether $f$ is linear or nonlinear

# Finite Difference Method, continued

- We replace derivatives by finite difference approximations such as

$$
\begin{aligned}
u'(t_i) &\approx \frac{y_{i+1} - y_{i-1}}{2h} && \longleftarrow \textit{Error is } O(h^2) \\
u''(t_i) &\approx \frac{y_{i+1} - 2y_i + y_{i-1}}{h^2} && \longleftarrow \textit{Error is } O(h^2)
\end{aligned}
$$

- This yields system of equations

$$\frac{y_{i+1} - 2y_i + y_{i-1}}{h^2} = f\left(t_i, y_i, \frac{y_{i+1} - y_{i-1}}{2h}\right)$$

  to be solved for unknowns $y_i$, $i = 1, \ldots, n$

- System of equations may be linear or nonlinear, depending on whether $f$ is linear or nonlinear

# Finite Difference Method, continued

- For these particular finite difference formulas, system to be solved is tridiagonal, which saves on both work and storage compared to general system of equations

- This is generally true of finite difference methods: they yield sparse systems because each equation involves few variables

# Example: Convection-Diffusion Equation

$$-\nu \frac{d^2u}{dx^2} \; + \; c\frac{du}{dx} \; = \; 1, \;\; u(0) \; = \; u(1) \; = \; 0,$$

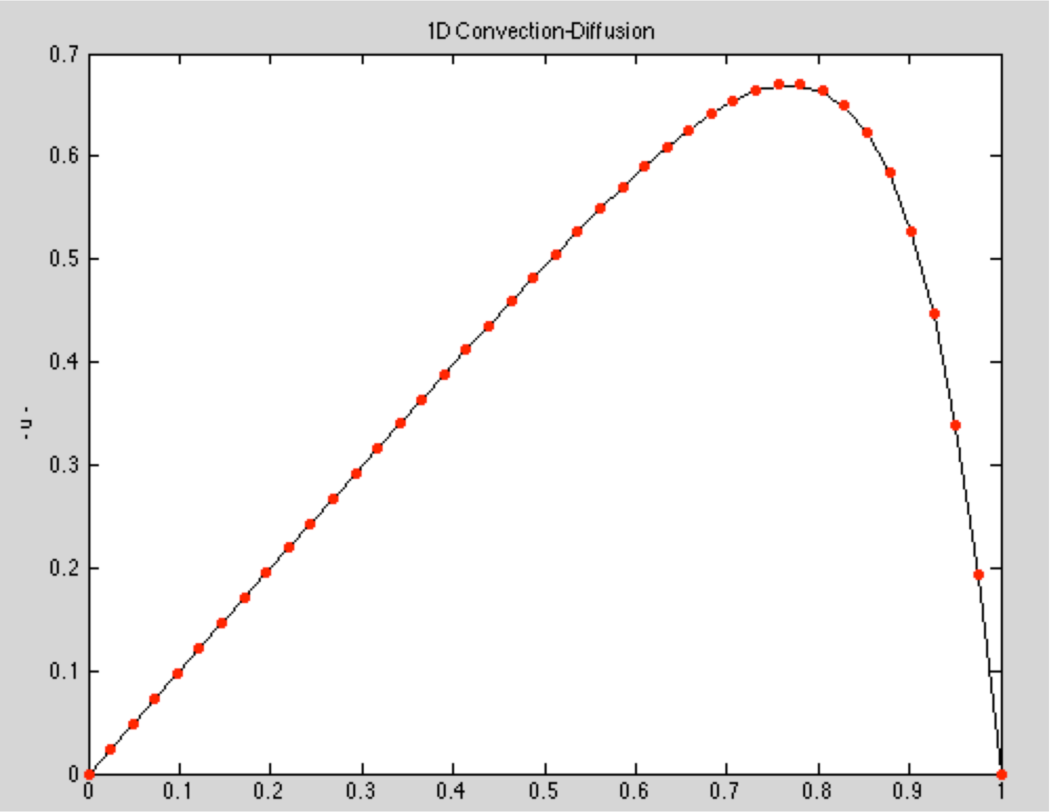
Apply finite difference: $L\mathbf{u} \; = \; A\mathbf{u} \; + \; C\mathbf{u} \; = \; \mathbf{f}$

$$A \; = \; \frac{\nu}{\Delta x^2} \begin{bmatrix} 2 & -1 & & & \\ -1 & 2 & \ddots & & \\ & \ddots & \ddots & \ddots & \\ & & \ddots & \ddots & -1 \\ & & & -1 & 2 \end{bmatrix} \qquad C \; = \; \frac{c}{2\Delta x} \begin{bmatrix} 0 & -1 & & & \\ -1 & 0 & \ddots & & \\ & \ddots & \ddots & \ddots & \\ & & \ddots & \ddots & -1 \\ & & & -1 & 0 \end{bmatrix}$$

*MATLAB EXAMPLE*

- $A$ is symmetric positive definite.
- $C$ is skew-symmetric.
- $L = A + C$ is neither SPD nor skew-symmetric.

# Example: Convection-Diffusion Equation



```
format compact

a = 1; b=-2; c=1;
e = ones(n,1);
A = spdiags([a*e b*e c*e],-1:1, n,n);

a =-1; b=0; c=1;
e = ones(n,1);
C = spdiags([a*e b*e c*e],-1:1, n,n);

h = 1./(n+1);

nu = .1; c=0.001;
nu = .01; c=1;
nu = .001; c=1;

A = -(nu/(h*h))*A;
C = (c/(2*h))*C;

L = A+C;

f = ones(n,1);

u = L\f;

x = (1:n)*h; x=x';
x=[ 0; x ; 1];
u=[ 0; u ; 0];

ue = (x - (exp(c*x/nu)-1)/(exp(c/nu)-1) )/c;
ue = (x - ( exp(c*(x-1)/nu)-exp(-c/nu) )/(1-exp(-c/nu)) )/c;

plot(x,ue,'k-',x,u,'r.-')
title('1D Convection-Diffusion')
xlabel(' - x - ')
ylabel(' - u - ')

norm(ue-u)/norm(ue)
```

# Comments About Computing Error Norms

- Be careful with the $l_2$ vector norm*!*

- Even though $\max |e_i| \longrightarrow 0$ with $n \longrightarrow \infty$, we can still have $||\mathbf{e}||$ grow with $n$. Why?

- When solving differential equations, it is better to use norms that approximate their continuous counterparts. Thus

$$\begin{aligned} ||e||_2 &= \left[ \int_\Omega e^2\, dx \right]^{\frac{1}{2}} \approx \left[ \frac{1}{n} \sum_{i=1}^{n} |e_i|^2 \right]^{\frac{1}{2}} \\ ||e||_\infty &= \max_\Omega |e| \approx \max_i |e_i| \end{aligned}$$

- The issue can also be resolved by measuring *relative error:*

$$error \; := \; \frac{||\mathbf{e}||}{||\mathbf{u}||}$$

  for some appropriate vector norm.

- Still, best to start with a norm that doesn't scale with $n$.

# Convection-Diffusion Equation

- Consider 1D convection-diffusion with $c = 1$ and $f = 1$:

$$\frac{\partial u}{\partial t} + c\,\frac{\partial u}{\partial x} = \nu \frac{\partial^2 u}{\partial x^2} + f$$

$$u(0) = 0, \quad u(1) = 0.$$

- Assume steady-state conditions $u_t = 0$

$$-\nu u_{xx} + c\,u_x = 1, \quad u(0) = u(1) = 0.$$

- If $\nu = 0$, we have:

$$c\,u_x = 1, \quad u(0) = u(1) = 0\ ???$$

  Too many boundary conditions!

# Convection-Diffusion Equation

- The issue is that $\nu \longrightarrow 0$ is a *singular perturbation.*
- This is true whenever the *highest-order derivative* is multiplied by a small constant.
- As the constant goes to zero, the number of boundary conditions changes.
- Here,
  - We go from one boundary condition when $\nu = 0$,
  - to two boundary conditions when $\nu > 0$ (even for $\nu \ll 1$).
- An example that is *not* a singular perturbation is

$$-u_{xx} + \epsilon\, u_x \;=\; 1, \quad u(0) \;=\; u(1) \;=\; 0, \qquad \epsilon \longrightarrow 0.$$

This is called a *regular perturbation.*

# Regular / Singular Perturbations You're Familiar With

- Consider solutions to the quadratic equation: $ax^2 + bx + c = 0.$

**Example 1:** $x^2 + \epsilon x = 1:$ Two roots as $\epsilon \longrightarrow 0$. *Regular perturbation.*

**Example 2:** $\epsilon x^2 + x = 1:$

$$\begin{aligned}
x &= -\frac{1}{2\epsilon} \pm \frac{1}{2\epsilon}\sqrt{1+4\epsilon} \\
x_1 &= \frac{1}{2\epsilon}\left(\sqrt{1+4\epsilon} - 1\right) \\
&= \frac{1}{2\epsilon}\left(1 + 2\epsilon - 1 + O(\epsilon^2)\right) \\
&= 1 + O(\epsilon). \\
x_2 &= -\frac{1}{2\epsilon}\left(2 + O(\epsilon)\right) \longrightarrow -\infty. \quad \textit{Singular perturbation.}
\end{aligned}$$

# Convection-Diffusion Equation

- Exact solution for our 1D model problem:

$$u = \frac{x}{c} - \frac{L}{c}\left[\frac{e^{cx/\nu} - 1}{e^{cL/\nu} - 1}\right]$$

$$= \frac{1}{c}\left[x - \frac{e^{c(x-L)/\nu} - e^{-cL/\nu}}{1 - e^{-cL/\nu}}\right].$$

- In the convection-dominated limit $(cL \gg \nu)$, one of these is computable in IEEE floating point, one is not.

- Which is which?

# Convection-Diffusion Equation

- What happens when cL/$\nu \gg$ 1 in our numerical example?

## Nonlinear Example: The Bratu Equation

- Consider 1D diffusion with nonlinear right-hand side:

$$-\frac{d^2u}{dx^2} \;=\; q(x,u) \;=\; \sigma\, e^u, \;\; u(0) = u(1) = 0.$$

- Discretizing with finite differences (say),

$$A\underline{u} = \sigma\, e^{\underline{u}}.$$

- Nonlinear system:

$$\underline{f}(\underline{u}) \;=\; 0, \quad \underline{f}(\underline{u}) \;=\; A\underline{u} \;-\; \sigma\, e^{\underline{u}}.$$

- Newton's method:

$$\begin{aligned} \underline{u}^{k+1} &= \underline{u}^{k} + \underline{s}^{k} \\ \underline{s}^{k} &= -J^{k}\,\underline{f}(\underline{u}^{k}). \end{aligned}$$

$$\left(J^{k}\right)_{ij} := \frac{\partial f_i^k}{\partial u_j^k}.$$

- $i$th equation:

$$\underline{f}_i^k = \sum_{j=1}^{n} A_{ij}\,u_j^k - \sigma e^{u_i^k} \qquad \longrightarrow \qquad (J_k)_{ij} = \frac{\partial f_i}{\partial u_j} = A_{ij} - \sigma e^{u_i}.$$

- If $b = -1$ and $a_j = 2 - h^2 \sigma e^{u_j}$, then

$$J = \frac{1}{h^2} \begin{pmatrix} a_1 & b & & & \\ b & a_2 & b & & \\ & b & \ddots & \ddots & \\ & & \ddots & \ddots & b \\ & & & b & a_n \end{pmatrix},$$

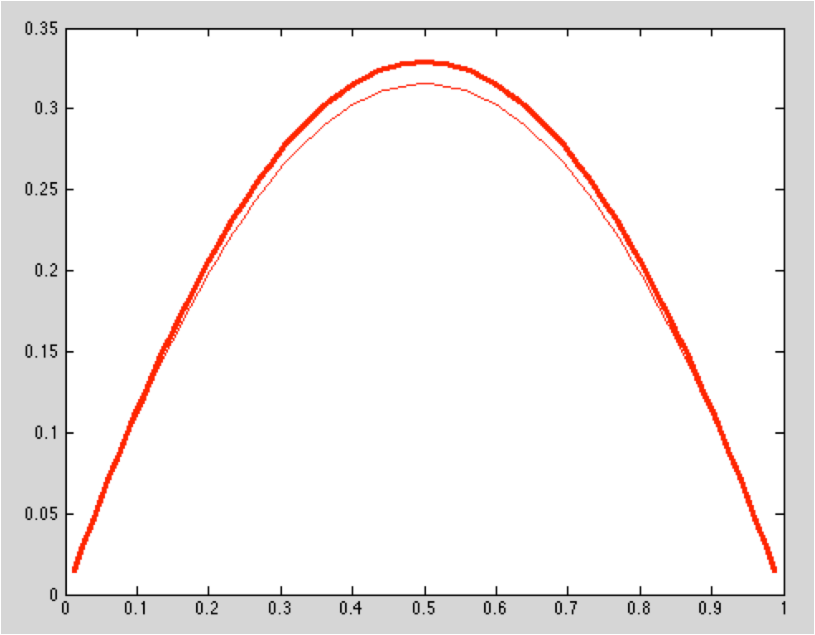
- At *each iteration*, modify the tridiagonal matrix $A$ such that

$$J_k = A + \sigma e^{u_i^k} \delta_{ij},$$

and solve this tridiagonal system in $\approx 8n$ operations.

# bratu1a.m

```
format compact; format longe;  close all

n=80; sigma = 2;

h=1./(n+1); b = ones(n,1); x=1:n; x=h*x'; h2i = 1./(h*h);
hold off; plot(x,0*x,'k-'); hold on

a=-2*b;
A = h2i*spdiags([b a b],-1:1, n,n);

c=-2*b + sigma*h*h*exp(x); J = h2i*spdiags([b c b],-1:1, n,n);


u=b*0;

for iter=1:31;

   f=A*u + sigma*exp(u);
   c=-2*b + sigma*h*h*exp(u);
   J = h2i*spdiags([b c b],-1:1, n,n);

   [L,U]=lu(J);

   s = -U\ (L\f);

   u = u+s;

   plot(x,u,'r-'); hold on

   ns = norm(s); nf = norm(f); [ns nf]

end;

plot(x,u,'r-'); hold on
```



# Extension of Finite Difference to Variable Coefficients

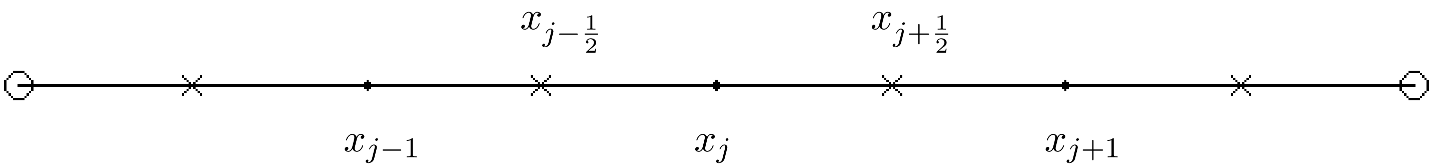


Figure 1: Grid spacing for variable coefficient diffusion operator.

Consider the one-dimesional model problem,

$$\frac{d}{dx}\, a(x) \frac{du}{dx} \;=\; f(x), \qquad u(0) \;=\; u(1) \;=\; 0. \tag{5}$$

Let

$$u_i \;:=\; u(x_i), \qquad a_{i+\frac{1}{2}} \;:=\; a(x_{i+\frac{1}{2}}). \tag{6}$$

with $x_i := i\,h$, $i = 0, \ldots, n+1$ and $x_{i+\frac{1}{2}} := (i+\frac{1}{2})\,h$, $i = 0, \ldots, n$, and $h := 1/(n+1)$. Then

$$w_i \;=\; \frac{d}{dx}\, a(x) \frac{du}{dx}\bigg|_{x_i} \;\approx\; \frac{1}{h}\left[ \left( a \frac{du}{dx} \right)\bigg|_{x_{i+\frac{1}{2}}} \;-\; \left( a \frac{du}{dx} \right)\bigg|_{x_{i-\frac{1}{2}}} \right] \tag{7}$$

$$\approx\; \frac{1}{h}\left[ a_{i+\frac{1}{2}} \left( \frac{u_{i+1} - u_i}{h} \right) \;-\; a_{i-\frac{1}{2}} \left( \frac{u_i - u_{i-1}}{h} \right) \right]. \tag{8}$$

# Extension of Finite Difference to Variable Coefficients

Assuming $u = 0$ at the domain endpoints, then the finite difference appoximation to $u'_{i+\frac{1}{2}}$, $i = 0, \ldots, n$ can be evaluated as the matrix-vector product, $\underline{v} = D\underline{u}$, where $D$ is the $(n+1) \times n$ finite difference matrix illustrated below.

$$\underline{v} \;=\; \begin{pmatrix} v_{\frac{1}{2}} \\ v_{\frac{3}{2}} \\ \vdots \\ v_{n+\frac{1}{2}} \end{pmatrix} \;=\; \frac{1}{h} \begin{bmatrix} 1 & & & \\ -1 & 1 & & \\ & -1 & \ddots & \\ & & \ddots & 1 \\ & & & -1 \end{bmatrix} \begin{pmatrix} u_1 \\ u_2 \\ \vdots \\ u_n \end{pmatrix} \;=\; D\underline{u}. \tag{9}$$

Note that $\frac{1}{h}(u_{i+1} - u_i)$ is generally regarded as a first-order accurate approximation to $\frac{du}{dx}$, it is in fact second-order accurate at the midpoint $x_{i+\frac{1}{2}}$.

Given $v_{i+\frac{1}{2}}$, it remains to evaluate the outer finite difference in (7), which maps data from the $(n+1)$ half-points to the $n$ integer points. Let

$$q_{i+\frac{1}{2}} \;:=\; a_{i+\frac{1}{2}} v_{i+\frac{1}{2}}. \tag{10}$$

Then

$$\underline{w} \;=\; \begin{pmatrix} w_1 \\ w_2 \\ \vdots \\ w_n \end{pmatrix} \;=\; \frac{1}{h} \begin{bmatrix} -1 & 1 & & & \\ & -1 & 1 & & \\ & & \ddots & \ddots & \\ & & & -1 & 1 \end{bmatrix} \begin{pmatrix} q_{\frac{1}{2}} \\ q_{\frac{3}{2}} \\ \vdots \\ q_{n+\frac{1}{2}} \end{pmatrix} \;=\; -D^T \underline{q}. \tag{11}$$

# Extension of Finite Difference to Variable Coefficients

Finally, note that if $A$ is an $(n+1) \times (n+1)$ diagonal matrix with entries $(a_{\frac{1}{2}},\, a_{\frac{3}{2}}, \ldots a_{n+\frac{1}{2}})$, then (10) can be expressed as $\underline{q} = A\underline{v}$, and the finite-difference approximation (7) can be compactly expressed in matrix form as

$$\underline{w} = -D^T AD\underline{u} \tag{12}$$

Assuming $a_{i+\frac{1}{2}} > 0$, it is easy to show that the matrix

$$L \quad := \quad D^T AD \tag{13}$$

is symmetric positive definite, which is a requirement if the system is to be solved with conjugate gradient iteration or Cholesky factorization. Fortunately, this property carries over into the multidimensional case, which we consider in the next section. We further remark that $L$ is a map from data $(u_1 \ldots u_j \ldots u_n)$ to $(w_1 \ldots w_j \ldots w_n)$. That is, once defined, it does not generate data at the half gridpoint locations. This is a particularly attractive feature in multiple space dimensions where having multiple grids leads to an explosion of notational difficulties.

# Convergence Behavior: Finite Difference

- In differential equations, we are interested in the rate of convergence – i.e., the rate at which the error goes to zero vs. n, the number of unknowns in the system.

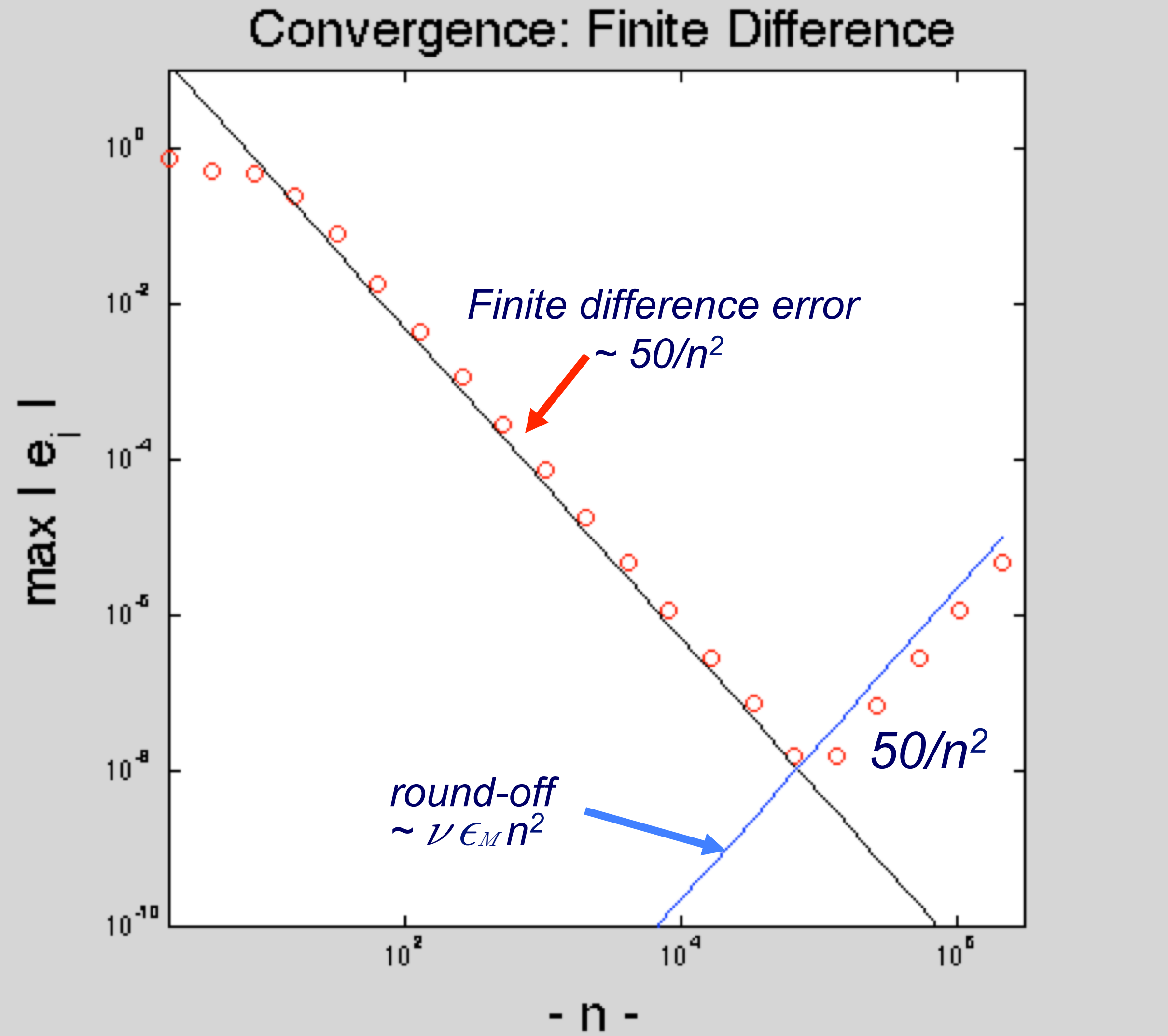
- For finite difference methods and methods using Lagrangian interpolants, n is the number of gridpoints (but, depends on the type of boundary conditions…..)

- The next figure shows the error vs. n for a 2nd-order (i.e., $O(h^2)$) finite difference solution of the steady-state convection-diffusion equation in 1D.

- For n > ~ $\epsilon_M^{-1/3}$, the error goes up, due to round-off associated with the approximation to the 2nd-order derivative.

- As we've seen in past homework assignments, the minimum error is around $\epsilon_M^{-1/2}$

# Finite Difference Convergence Rate



*Finite difference error*
*~ $50/n^2$*

*round-off*
*~ $\nu \epsilon_M n^2$*

*$50/n^2$*

# Properties of Finite Difference Methods

- ❑ Pros
  - ❑ Easy to formulate (for simple problems)
  - ❑ Easy to analyze "
  - ❑ Easy to code "
  - ❑ Closed-form expressions for eigenvalues/eigenvectors for uniform grid with constant coefficients.

- ❑ Cons –
  - ❑ Geometric complexity for 2D/3D is not as readily handled as FEM.
  - ❑ Difficult to extend to high-order (because of boundary conditions).
  - ❑ Do not always (e.g., because of BCs) get a symmetric matrix for

$$\frac{d^2u}{dx^2} \qquad \text{or} \qquad \frac{d}{dx}\nu(x)\frac{du}{dx}$$

# Eigenvalues, Continuous and Discrete

❑ One of the great features of finite difference methods is that one can readily compute the eigenvalues of the discrete operators and thus understand their spectrum and convergence rates.

❑ The latter is important for understanding accuracy.

❑ The former is important for understanding stability of time-stepping schemes in the case of PDEs, which we'll see in the next chapter.

❑ The reason it is easy to find the eigenvalues for finite difference methods is that, for the constant coefficient case, they often share the same eigenfunctions as their continuous counterparts.

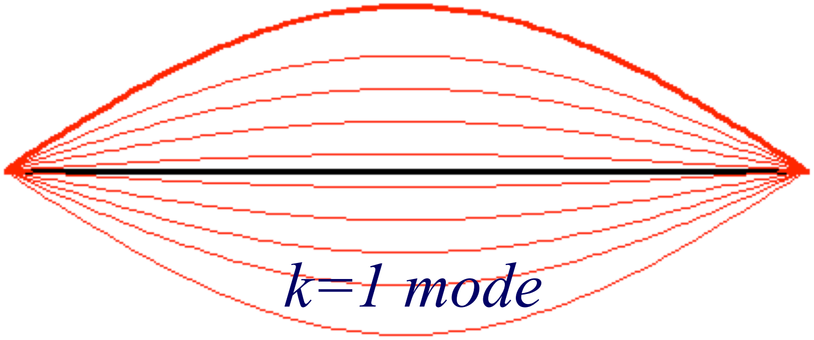
# Eigenvalue Example:

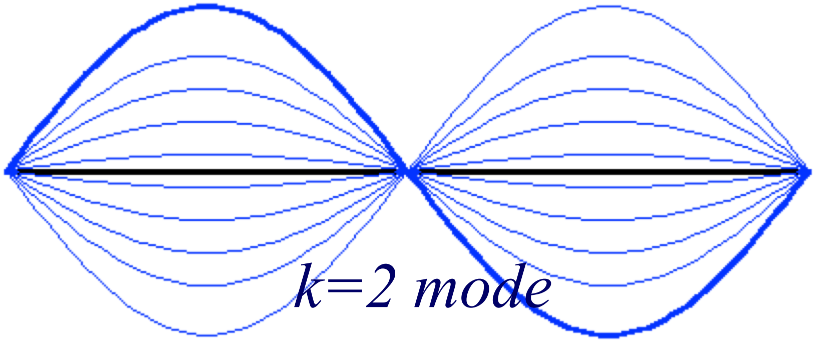
- Consider the analytical (i.e., continuous) eigenvalue problem

$$-\frac{d^2\tilde{u}}{dx^2} \;=\; \tilde{\lambda}\,\tilde{u}, \qquad \tilde{u}(0) = \tilde{u}(1) = 0.$$

- The eigenfunctions/eigenvalues for the continuous problem are

$$\begin{aligned} \tilde{u} &= \sin(k\pi x) : \\ & \qquad -\tilde{u}'' \;=\; k^2\pi^2 \sin(k\pi x) \;=\; k^2\pi^2\,\tilde{u} \;=\; \tilde{\lambda}_k\,\tilde{u} \\ \tilde{\lambda}_k &= k^2\pi^2 \end{aligned}$$





The modes are like the vibrations of a guitar string.
Higher wavenumbers, k, correspond to higher frequency.
Here, the k=2 mode would be a harmonic – one octave higher.

## Finite Difference Eigenvectors/values:

- Consider $\mathbf{s} = [\sin(k\pi x_j)]^T$:

$$\begin{aligned} A\mathbf{s}|_j &= \frac{-1}{\Delta x^2}\left[\sin k\pi x_{j+1} \;-\; 2\sin k\pi x_j \;+\; \sin k\pi x_{j-1}\right] \\ &= \frac{-1}{\Delta x^2}\left[\sin(k\pi x_j + \Delta x) \;-\; 2\sin(k\pi x_j) \;+\; \sin(k\pi x_j - \Delta x)\right] \end{aligned}$$

- Use the identity:

$$\sin(a+b) \;=\; \sin a\,\cos b \;+\; \cos a\,\sin b$$

$$\begin{aligned} \sin(k\pi x_{j+1}) &= \sin k\pi x_j\;\cos k\pi\Delta x \;+\; \cos k\pi x_j\;\sin k\pi\Delta x \\ \sin(k\pi x_{j-1}) &= \sin k\pi x_j\;\cos k\pi\Delta x \;-\; \cos k\pi x_j\;\sin k\pi\Delta x \\ \textsc{sum} &= 2\;\sin k\pi x_j\;\cos k\pi\Delta x \end{aligned}$$

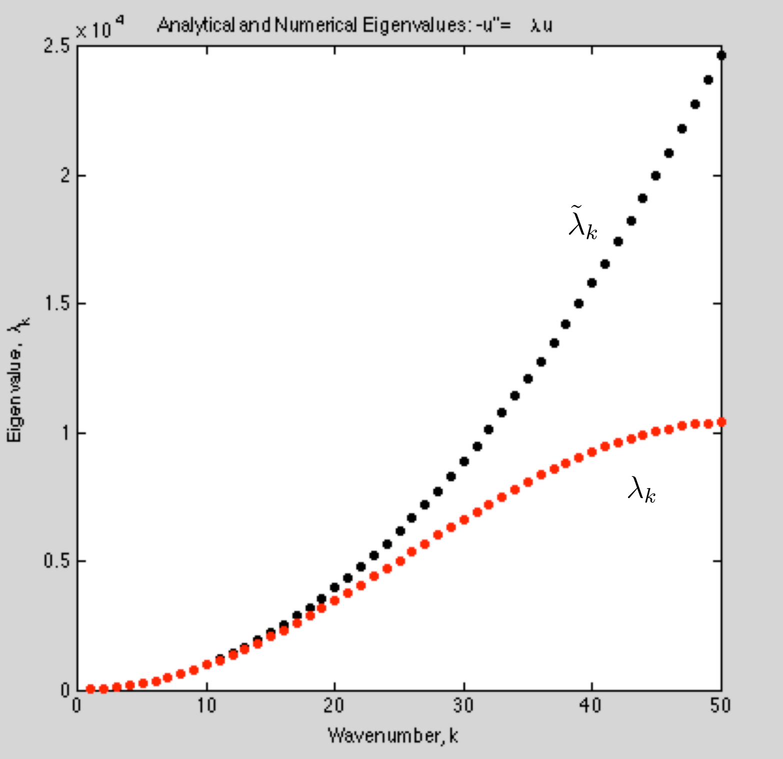
- $$\begin{aligned} A\mathbf{s}|_j = \frac{-1}{\Delta x^2}\left[s_{j+1} \;-\; 2s_j \;+\; s_{j-1}\right] &= -\frac{1}{\Delta x^2}\left[2\cos k\pi\Delta x - 2\right]\sin k\pi x_j \\ &= \lambda_k \mathbf{s}|_j \end{aligned}$$

$$\boxed{\lambda_k \;=\; \frac{2}{\Delta x^2}\left[1 - \cos k\pi\Delta x\right].}$$

## Eigenvalue Properties for $-u'' = \lambda u$, $u(0) = u(1) = 0$:

- $\max \lambda_k \sim Cn^2$ $\qquad\qquad \dfrac{\tilde{\lambda}_n}{\lambda_n} \sim \dfrac{\pi^2}{4}$
- For $k\Delta x \ll 1$, (with $\theta := k\Delta x$):

$$\lambda_k \quad = \quad (k\pi)^2 \left[ 1 - \frac{(k\pi\Delta x)^2}{12} + \cdots \right]$$



# Collocation Method

- *Collocation method* approximates solution to BVP by finite linear combination of basis functions
- For two-point BVP

$$u'' = f(t, u, u'), \qquad a < t < b$$

with BC

$$u(a) = \alpha, \qquad u(b) = \beta$$

we seek approximate solution of form

$$u(t) \approx v(t, \boldsymbol{x}) = \sum_{i=1}^{n} x_i \phi_i(t)$$

where $\phi_i$ are basis functions defined on $[a, b]$ and $\boldsymbol{x}$ is $n$-vector of parameters to be determined

# Collocation Method, continued

- To determine vector of parameters $\boldsymbol{x}$, define set of $n$ *collocation points*, $a = t_1 < \cdots < t_n = b$, at which approximate solution $v(t, \boldsymbol{x})$ is forced to satisfy ODE and boundary conditions

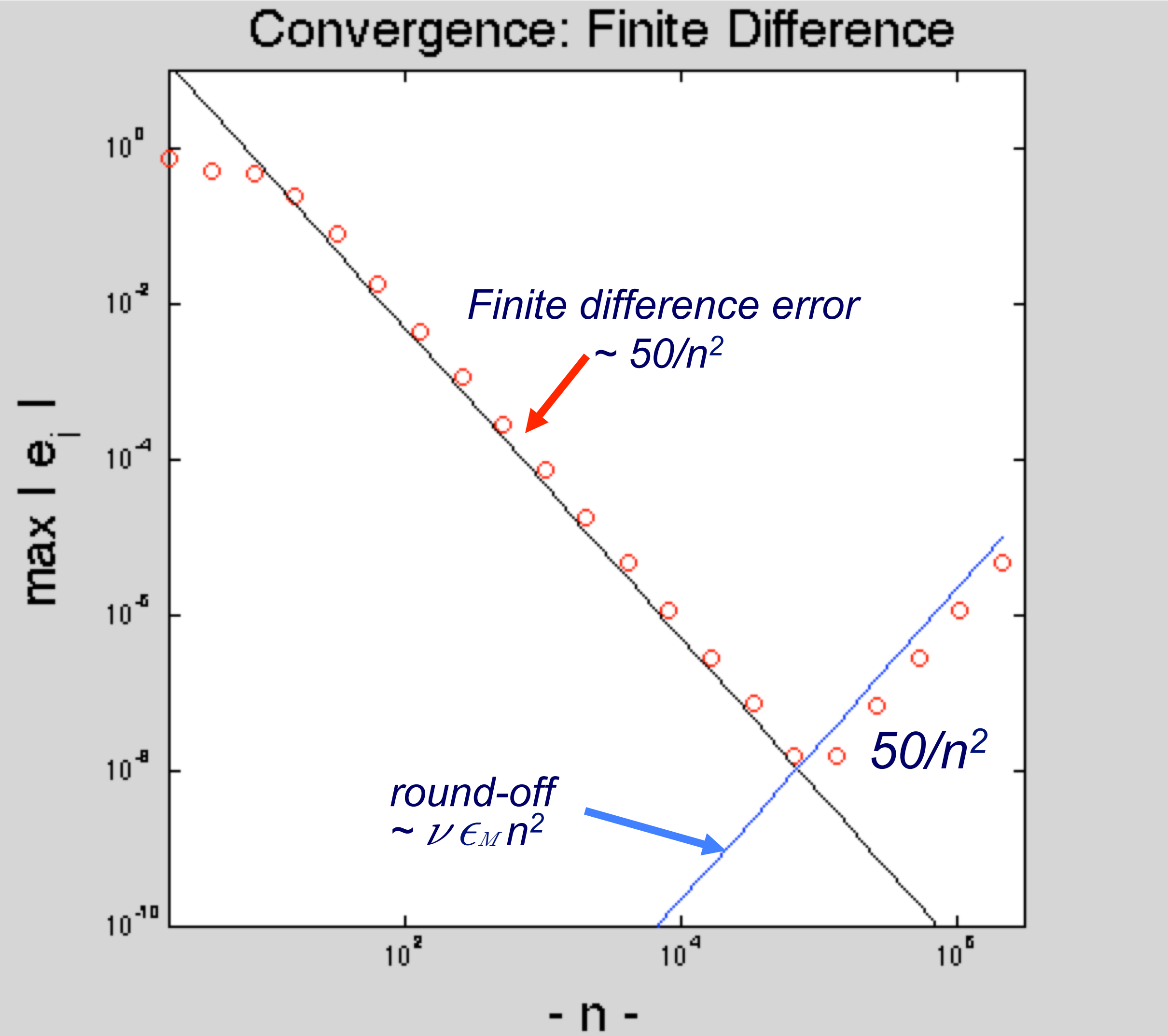
- Common choices of collocation points include equally-spaced points or Chebyshev points

- Suitably smooth basis functions can be differentiated analytically, so that approximate solution and its derivatives can be substituted into ODE and BC to obtain system of algebraic equations for unknown parameters $\boldsymbol{x}$

# Collocation

- Collocation is essentially the ***method of undetermined coefficients***.
  - Start with
    $$u(x) = \sum_{j=0}^{n} \hat{u}_j \phi_j(x)$$
    .
  - Find coefficients $\hat{u}_j$ such that BVP is satisfied at gridpoints $x_i$.
- Instead of using monomials, $\phi_j = x^j$, could use Lagrange polynomials on Chebyshev or Legendre quadrature points.
- Normally, one would use Gauss-Lobatto-Legendre or Gauss-Lobatto-Chebyshev points, which include $\pm 1$ (i.e., the endpoints of the interval) in the nodal set.
- If the solution is to be zero at those boundaries one would have $u_0 = u_n = 0$.
- In many cases, these methods are exponentially convergent. (**Counter-example:** sines and cosines, unless problem is periodic.)
- For several reasons  conditioning, symmetry, robustness, and ease of boundary condtions, collocation has lost favor to Galerkin methods.

# Finite Difference Convergence Rate



*Finite difference error*
*~ $50/n^2$*

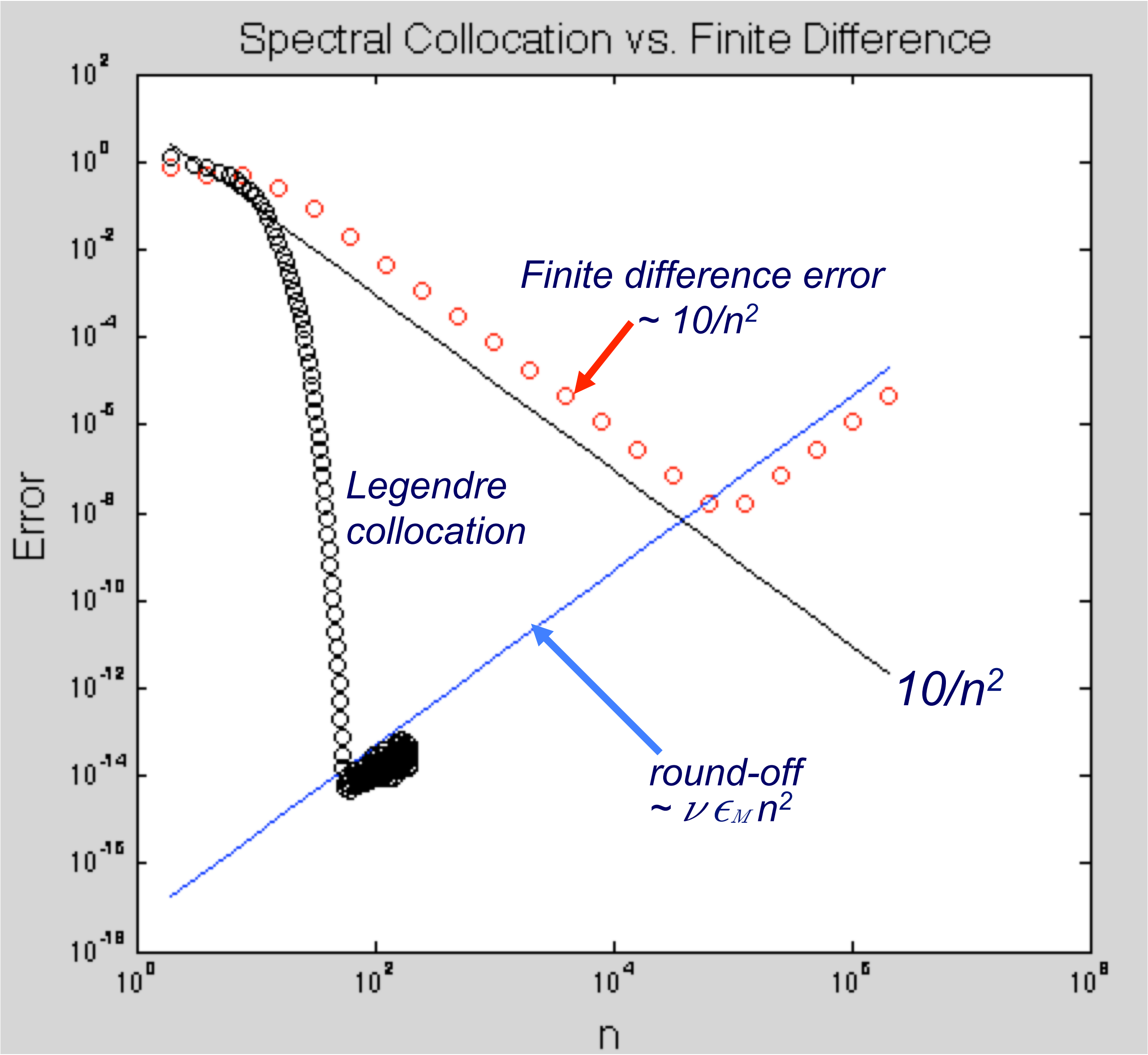
$50/n^2$

*round-off*
*~ $\nu \epsilon_M n^2$*

# Convergence Behavior: High-Order Methods

- ❑ The 2nd-order convergence of standard finite difference methods looks reasonable.
- ❑ However, higher-order methods are generally much faster in that the same error can be achieved with a lower n, once n is large enough for the asymptotic convergence behavior to apply.
- ❑ High-order methods suffer the same round-off issue, with error growing like $\epsilon_M n^2$.
- ❑ However, their truncation error goes to zero more rapidly so that the n where truncation and round-off errors balance is lower and the minimum error is thus much smaller.
- ❑ Usually, we are more interested in a small error at small n, rather than realizing the minimum possible error.
- ❑ For PDEs on an (n x n x n) grid cost generally scales as $n^3$, so a small n is a significant win.

# Spectral Collocation vs. Finite Difference



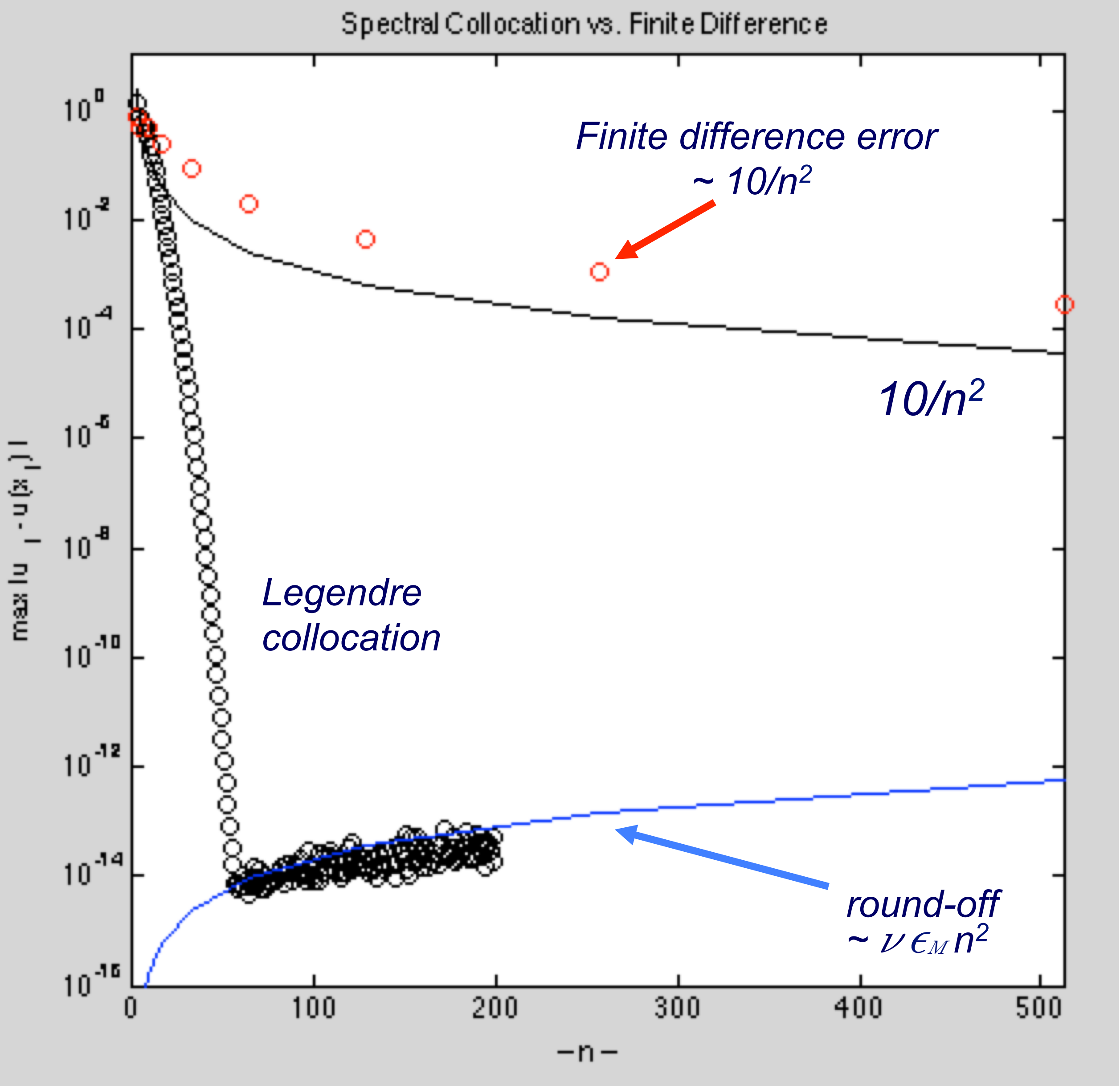
*Finite difference error*
*~ $10/n^2$*

*Legendre collocation*

*$10/n^2$*

*round-off*
*~ $\nu \, \epsilon_M \, n^2$*

# Spectral Collocation vs. Finite Difference (semilogy)



*Finite difference error* $\sim 10/n^2$

$10/n^2$

*Legendre collocation*

*round-off* $\sim \nu \epsilon_M n^2$

# Galerkin Method

- Rather than forcing residual to be zero at finite number of points, as in collocation, we could instead minimize residual over entire interval of integration
- For example, for *Poisson equation* in one dimension,

$$u'' = f(t), \qquad a < t < b,$$

with homogeneous BC $\quad u(a) = 0, \qquad u(b) = 0,$

subsitute approx solution $\quad u(t) \approx v(t, \boldsymbol{x}) = \sum_{i=1}^{n} x_i \phi_i(t)$

into ODE and define residual

$$r(t, \boldsymbol{x}) = v''(t, \boldsymbol{x}) - f(t) = \sum_{i=1}^{n} x_i \phi_i''(t) - f(t)$$

# Galerkin Method, continued

- More generally, *weighted residual method* forces residual to be orthogonal to each of set of *weight functions* or *test functions* $w_i$,

$$\int_a^b r(t, \boldsymbol{x}) w_i(t)\, dt = 0, \quad i = 1, \ldots, n$$

- This yields linear system $\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}$, where now

$$a_{ij} = \int_a^b \phi_j''(t) w_i(t)\, dt, \qquad b_i = \int_a^b f(t) w_i(t)\, dt$$

whose solution gives vector of parameters $\boldsymbol{x}$

# Galerkin Method, continued

- Matrix resulting from weighted residual method is generally not symmetric, and its entries involve second derivatives of basis functions
- Both drawbacks are overcome by *Galerkin method*, in which weight functions are chosen to be *same* as basis functions, i.e., $w_i = \phi_i$, $i = 1, \dots, n$
- Orthogonality condition then becomes

$$\int_a^b r(t, \boldsymbol{x}) \phi_i(t)\, dt = 0, \quad i = 1, \dots, n$$

or

$$\int_a^b v''(t, \boldsymbol{x}) \phi_i(t)\, dt = \int_a^b f(t) \phi_i(t)\, dt, \quad i = 1, \dots, n$$

# Galerkin Method, continued

- Degree of differentiability required can be reduced using integration by parts, which gives

$$\begin{aligned}\int_a^b v''(t,\boldsymbol{x})\phi_i(t)\,dt &= v'(t)\phi_i(t)|_a^b - \int_a^b v'(t)\phi_i'(t)\,dt \\ &= v'(b)\phi_i(b) - v'(a)\phi_i(a) - \int_a^b v'(t)\phi_i'(t)\,dt\end{aligned}$$

- Assuming basis functions $\phi_i$ satisfy homogeneous boundary conditions, so $\phi_i(0) = \phi_i(1) = 0$, orthogonality condition then becomes

$$-\int_a^b v'(t)\phi_i'(t)\,dt = \int_a^b f(t)\phi_i(t)\,dt, \quad i = 1,\ldots,n$$

# Galerkin Method, continued

- This yields system of linear equations $\boldsymbol{Ax} = \boldsymbol{b}$, with

$$a_{ij} = -\int_a^b \phi_j'(t)\phi_i'(t)\,dt, \qquad b_i = \int_a^b f(t)\phi_i(t)\,dt$$

  whose solution gives vector of parameters $\boldsymbol{x}$

- $\boldsymbol{A}$ is symmetric and involves only first derivatives of basis functions

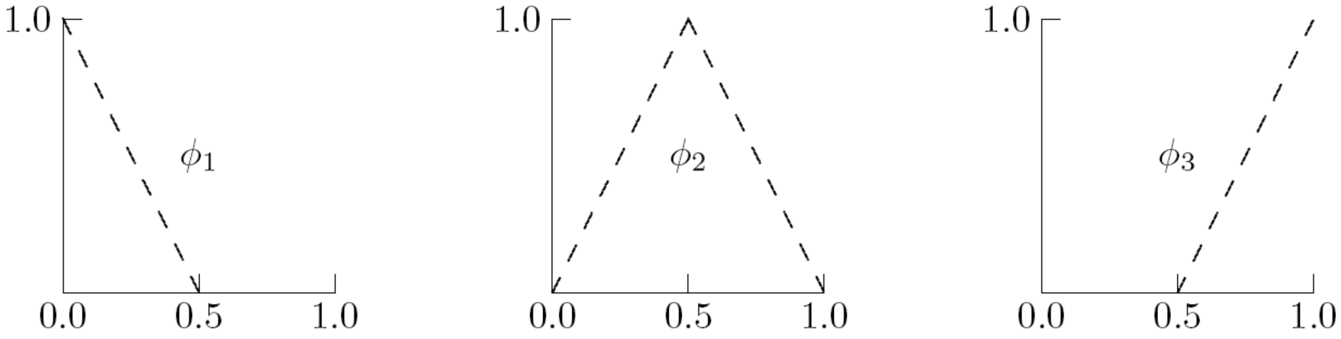
# Example: Galerkin Method

- Consider again two-point BVP

$$u'' = 6t, \qquad 0 < t < 1,$$

with BC $\quad u(0) = 0, \qquad u(1) = 1$

- We will approximate solution by piecewise linear polynomial, for which B-splines of degree $1$ (“hat” functions) form suitable set of basis functions



- To keep computation to minimum, we again use same three mesh points, but now they become knots in piecewise linear polynomial approximation

# Example, continued

- Thus, we seek approximate solution of form

$$u(t) \approx v(t, \boldsymbol{x}) = x_1\phi_1(t) + x_2\phi_2(t) + x_3\phi_3(t)$$

- From BC, we must have $x_1 = 0$ and $x_3 = 1$
- To determine remaining parameter $x_2$, we impose Galerkin orthogonality condition on interior basis function $\phi_2$ and obtain equation

$$-\sum_{j=1}^{3}\left(\int_0^1 \phi_j'(t)\phi_2'(t)\,dt\right)x_j = \int_0^1 6t\phi_2(t)\,dt$$

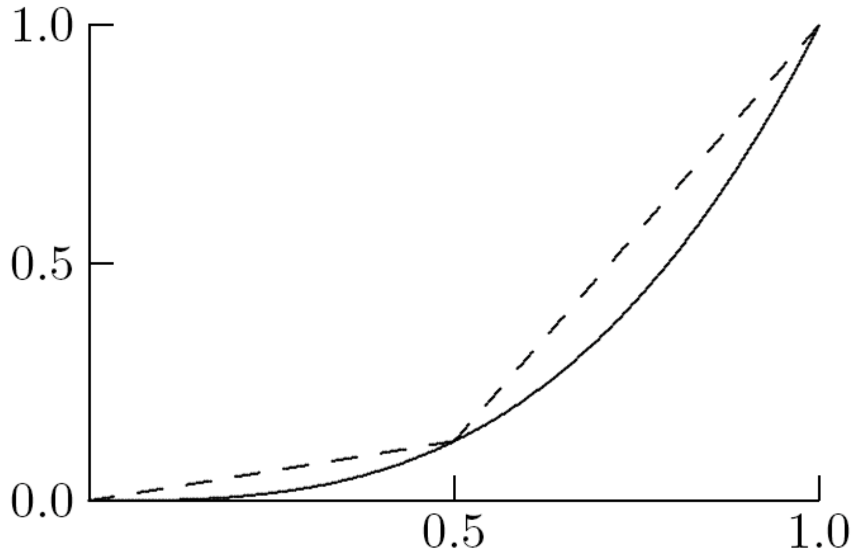
or, upon evaluating these simple integrals analytically

$$2x_1 - 4x_2 + 2x_3 = 3/2$$

# Example, continued

- Substituting known values for $x_1$ and $x_3$ then gives $x_2 = 1/8$ for remaining unknown parameter, so piecewise linear approximate solution is

$$u(t) \approx v(t, \boldsymbol{x}) = 0.125\phi_2(t) + \phi_3(t)$$



- We note that $v(0.5, \boldsymbol{x}) = 0.125$

# Example, continued

- More realistic problem would have many more interior mesh points and basis functions, and correspondingly many parameters to be determined

- Resulting system of equations would be much larger but still sparse, and therefore relatively easy to solve, provided local basis functions, such as “hat” functions, are used

- Resulting approximate solution function is less smooth than true solution, but nevertheless becomes more accurate as more mesh points are used

< interactive example >

# Weighted Residual Techniques (Galerkin, FEM, ...)

- Consider $\mathcal{L}\tilde{u} = f$, plus boundary conditions (e.g., $\mathcal{L}\tilde{u} = -\nu \tilde{u}_{xx} + c\tilde{u}_x$).
- Recall collocation with Lagrangian basis functions $\phi_j(x_i) = \delta_{ij}$:

$$\begin{aligned} u(x) &:= \sum_{j=0}^{n} u_j \phi_j(x) \\ r(x) &:= f - \mathcal{L}u \\ r(x_i) &:= f - \mathcal{L}u \\ &\Longrightarrow f(x_i) - [-\nu u_{xx} + c u_x]_{x_i} = 0 \end{aligned}$$

  - Unknown basis coefficients, $u_j$.
  - Residual set to zero, point-wise, at interior points $x_i$ .
  - Other two equations come from BCs.

- For *weighted residual technique* (WRT), we insist that the residual be orthogonal to a family of test functions,

$$\textit{Find } u \in X_0^N \textit{ such that } \int_a^b v\,(f - \mathcal{L}u)\,dx \;=\; \int_a^b v\,r\,dx \;=\; 0 \;\forall v\, \in Y_0^N$$

- Here, $X_0^N := \text{span}\{\phi_j\}$ is the space of *trial functions* and $Y_0^N := \text{span}\{\psi_i\}$ is the space of *test functions.*

- The relationship between collocation and WRT is analogous to the relationship between least squares and interpolation.

- One seeks a polynomial interpolant of degree $n$ (say) and chooses it either in a weighted (i.e., LS / WRT ) or pointwise ( interpolation / collocation ) fashion.

- Two common choices for $Y_0^N$ are

  - $\psi_i \;=\; \delta(x - x_i)$, the Dirac delta function, or
  - $\psi_i \;=\; \phi_i$, (i.e., $Y_0^N = X_0^N$), termed the Galerkin method.

- Galerkin is far more common.

- Dirac delta functions yield collocation, so we see that collocation is in fact a WRT.

- We'll focus solely on the Galerkin case.
- Most common bases for $X_0^N$ are
  - Lagrange polynomials (on *good* point sets)
  - Piecewise polynomials (FEM) – typically still Lagrangian

Questions that arise:

- Implementation
- Cost (strongly dependent on number of space dimensions)
- Accuracy
- Spectrum
- Other (e.g., optimality)

Let's consider the 1D Poisson equation

$$-\frac{d^2\tilde{u}}{dx^2} \;=\; f(x), \qquad \tilde{u}(0) \;=\; 0, \;\; \left.\frac{d\tilde{u}}{dx}\right|_{x=1} \;=\; 0.$$

We seek an approximate solution $u$ from a finite-dimensional *trial space* $X_0^N$,

$$u \in X_0^N \;:=\; \text{span}\{\phi_1(x),\, \phi_2(x), \dots,\, \phi_n(x)\}, \qquad \phi_j(0) = 0.$$

(We use the subscript on $X_0^N$ to indicate that functions in this space satisfy the homogeneous Dirichlet boundary conditions.)

# Trial Solution and Residual

The *trial solution* has the form

$$\boxed{u(x) \;=\; \sum_{j=1}^{n} \phi_j(x)\hat{u}_j .}$$

The $\phi_j$'s are the *basis functions.*

The $\hat{u}_j$'s are the *basis coefficients.*

We define the *residual*, $r(x;u) = r(x)$, as

$$r(x) \;:=\; f(x) \,+\, \frac{d^2u}{dx^2}.$$

It is clear that $r$ is some measure of the error given that

$$r \;\equiv\; 0 \quad \textit{iff} \quad u \;=\; \tilde{u}.$$

(In fact, it is the *only* measure of error available to us.)

# Trial Solution and Residual

Another equivalent definition of the residual derives from the fact that

$$f(x) \equiv -\frac{d^2\tilde{u}}{dx^2}$$

for the exact solution $\tilde{u}(x)$. Substituting, we have,

$$r(x) := f(x) + \frac{d^2u}{dx^2} = -\frac{d^2}{dx^2}\left(\tilde{u} - u\right) = -\frac{d^2e}{dx^2},$$

where $e(x) := \tilde{u}(x) - u(x)$ is the *error*.

The residual associated with $u(x)$ is thus the differential operator applied to the error function (with homogeneous boundary conditions).

This form will be of value later on.

# WRT and Test Functions

In the WRT, we don't require $r \equiv 0$.

Rather, we insist that $r$ be ($\mathcal{L}^2$-) orthogonal to a set of functions $v$ belonging to the the *test space*, $Y_0^N$,

$$\int_0^1 v\, r\, dx \;=\; 0, \qquad \forall\, v \in Y_0^N.$$

Convergence is attained as we complete the approximation space, that is, as we let $n \longrightarrow \infty$ for a reasonable set of $\phi_j$s.

It is most common to take the trial and test spaces to be the same, $Y_0^N = X_0^N$, which leads to the *Galerkin* formulation,

$$\textit{Find } u \in X_0^N \textit{ such that}$$

$$-\int_0^1 v\, \frac{d^2u}{dx^2}\, dx \;=\; \int_0^1 v\, f\, dx \qquad \forall\, v \in X_0^N.$$

# Reducing Continuity to $C^0$

It appears that $u$ must be twice differentiable.

However, if we integrate by parts, we can reduce the continuity requirements on $u$.

Let $\mathcal{I}$ denote the l.h.s. of the preceding equation:

$$\begin{aligned}
\mathcal{I} &= -\int_0^1 v\,\frac{d^2u}{dx^2}\,dx \\
&= \int_0^1 \frac{dv}{dx}\,\frac{du}{dx}\,dx \;-\; \cancel{v\,\frac{du}{dx}\bigg|_0^1} \\
&= \int_0^1 \frac{dv}{dx}\,\frac{du}{dx}\,dx
\end{aligned}$$

For a variety of technical reasons, it's generally a good idea to balance the continuity requirements of $v$ and $u$, to the extent possible.

# Weighted Residual / Variational Formulation

Using the integration-by-parts trick of the preceding slide (the only bit of calculus we'll require), we arrive at the weighted residual statement for $u$.

*Find* $u \in X_0^N$ *such that*

$$\int_0^1 \frac{dv}{dx}\frac{du}{dx}\,dx \;=\; \int_0^1 v\,f\,dx \qquad \forall\, v \in X_0^N.$$

Convergence is attained by taking the limit $n \longrightarrow \infty$ for an appropriate set of basis functions in $X_0^N$.

# Formulation of the Discrete Problem

We can now easily generate our discrete system that allows us to compute the set of basis coefficients. Let

$$\begin{aligned} \underline{u} &:= (u_1\, u_2\, \ldots\, u_n)^T, \\ \underline{v} &:= (v_1\, v_2\, \ldots\, v_n)^T. \end{aligned}$$

Then

$$\begin{aligned} \mathcal{I} := \int_\Omega v' u' \, dx &= \int_\Omega \left( \sum_{i=1}^n \phi_i'(x) v_i \right) \left( \sum_{j=1}^n \phi_j'(x) u_j \right) dx \\ &= \sum_{i=1}^n \sum_{j=1}^n v_i \left( \int_\Omega \phi_j'(x) \phi_i'(x) \, dx \right) u_j \\ &= \sum_{i=1}^n \sum_{j=1}^n v_i \, A_{ij} \, u_j, \;=\; \underline{v}^T A \underline{u}, \end{aligned}$$

with the (global) *stiffness matrix*, $A$, given by

$$\boxed{A_{ij} \;:=\; \int_\Omega \phi_j'(x) \phi_i'(x) \, dx.}$$

# Formulation of the Discrete Problem

We proceed in a similar way with the right-hand side. Assuming

$$f(x) \quad = \quad \sum_{j=0}^{n} \phi_j(x) f_j$$

(which is *way* overly restrictive, since $f \in \mathcal{L}^2_\Omega$ suffices), then

$$\begin{aligned}
\mathcal{I} \;=\; \int_\Omega v f \, dx \quad &= \quad \left( \sum_{i=1}^{n} \phi_i(x) v_i \right) \left( \sum_{j=1}^{n} \phi_j(x) f_j \right) \, dx \\
&= \quad \sum_{i=1}^{n} \sum_{j=1}^{n} v_i \, \left( \int_\Omega \phi_i(x) \phi_j(x) \, dx \right) \, f_j \\
&= \quad \sum_{i=1}^{n} \sum_{j=1}^{n} v_i \, B_{ij} \, f_j, \;=\; \underline{v}^T B \underline{f},
\end{aligned}$$

with the (global) *mass matrix*, $B$, given by

$$B_{ij} \quad := \quad \int_\Omega \phi_i(x) \phi_j(x) \, dx.$$

# Formulation of the Discrete Problem

Combining the results of the two previous slides, we have:

$$\mathcal{I} \;=\; \underline{v}^T A \underline{u} = \underline{v}^T B \underline{f} \qquad \forall \underline{v} \in \mathbb{R}^n,$$

which implies

$$A\underline{u} = B\underline{f}.$$

Since $A$ is symmetric positive definite, this system is *solvable*.

# Choice of Spaces & Bases

- *The next step is to choose the* ***space****,* $X_0^N$*, and associated* ***basis*** $\{ \phi_i \}$.
- The former influences convergence, i.e.,
  - How large or small n must be for a given error.
- The latter influences implementation, i.e.,
  - details and level of complexity, and
  - performance (time to solution, for a given error)

# Unstable and Stable Bases within the Elements

- *Examples of unstable bases are:*
  - Monomials (modal): $\phi_i = x^i$
  - High-order Lagrange interpolants (nodal) on *uniformly-spaced* points.

- Examples of *stable* bases are:
  - Orthogonal polynomials (modal), e.g.,
    - Legendre polynomials: $L_k(x)$, or
    - bubble functions: $\phi_k(x) := L_{k+1}(x) – L_{k-1}(x)$.
  - Lagrange (nodal) polynomials based on Gauss quadrature points (e.g., Gauss-Legendre, Gauss-Chebyshev, Gauss-Lobatto-Legendre, etc.)

# Piecewise Polynomial Bases: Linear and Quadratic

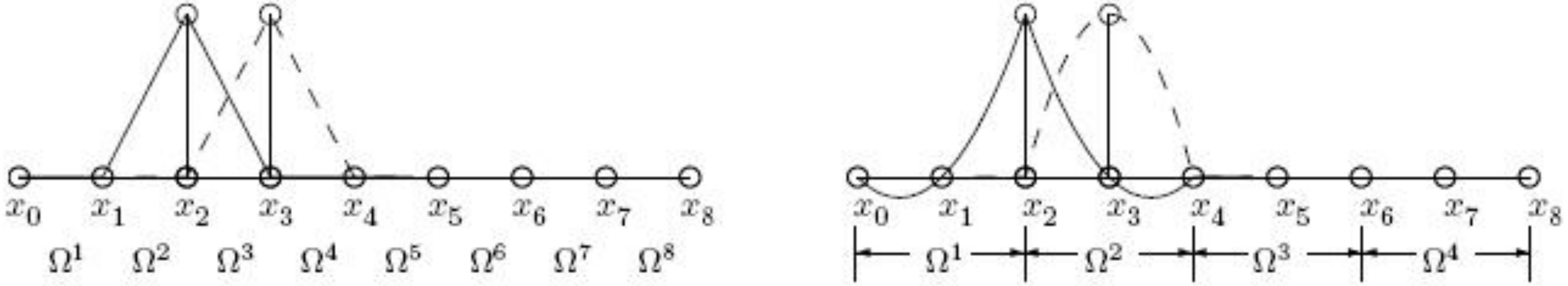
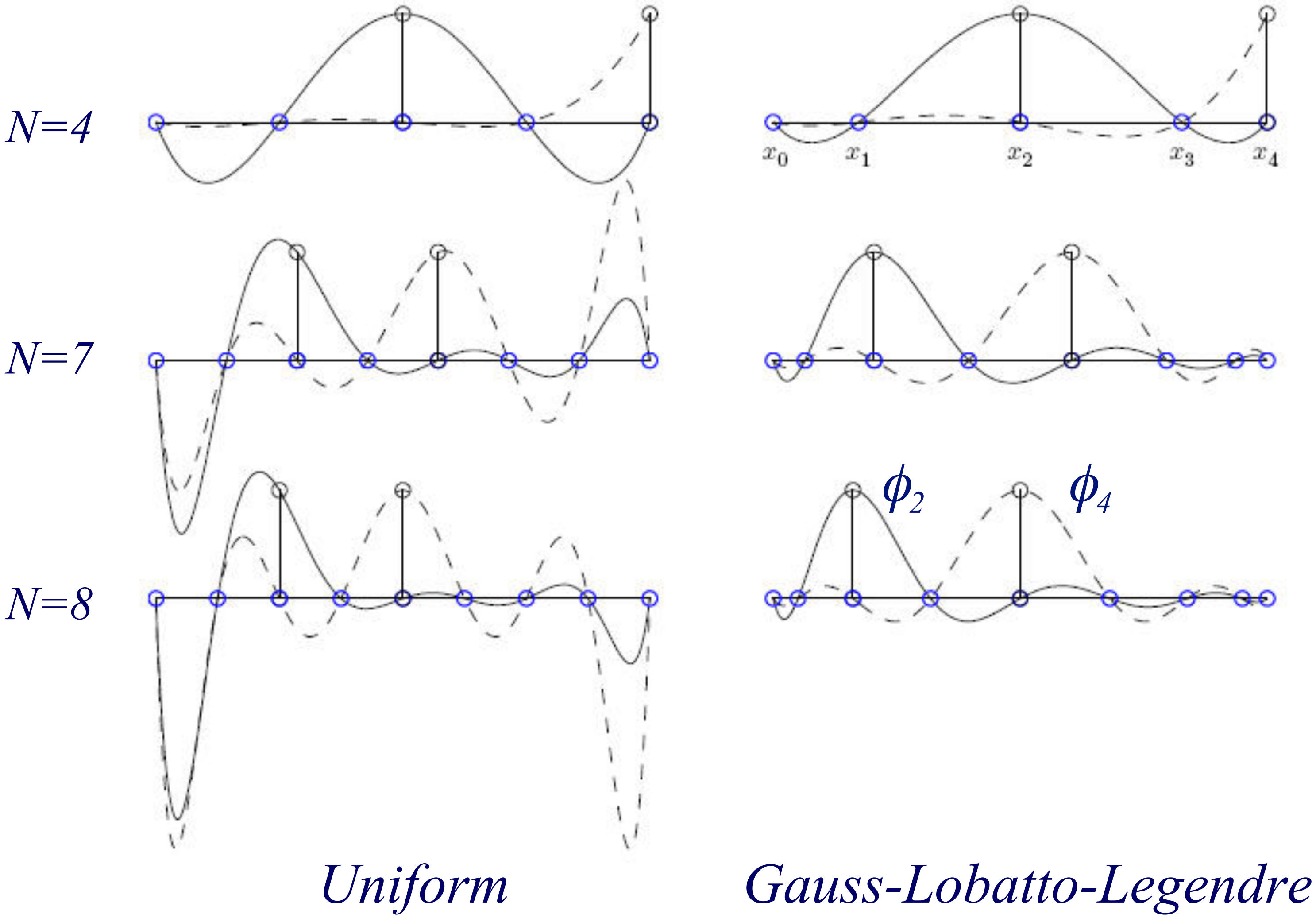

$x_0$ $x_1$ $x_2$ $x_3$ $x_4$ $x_5$ $x_6$ $x_7$ $x_8$
$\Omega^1$ $\Omega^2$ $\Omega^3$ $\Omega^4$ $\Omega^5$ $\Omega^6$ $\Omega^7$ $\Omega^8$
$x_0$ $x_1$ $x_2$ $x_3$ $x_4$ $x_5$ $x_6$ $x_7$ $x_8$
$\Omega^1$ $\Omega^2$ $\Omega^3$ $\Omega^4$

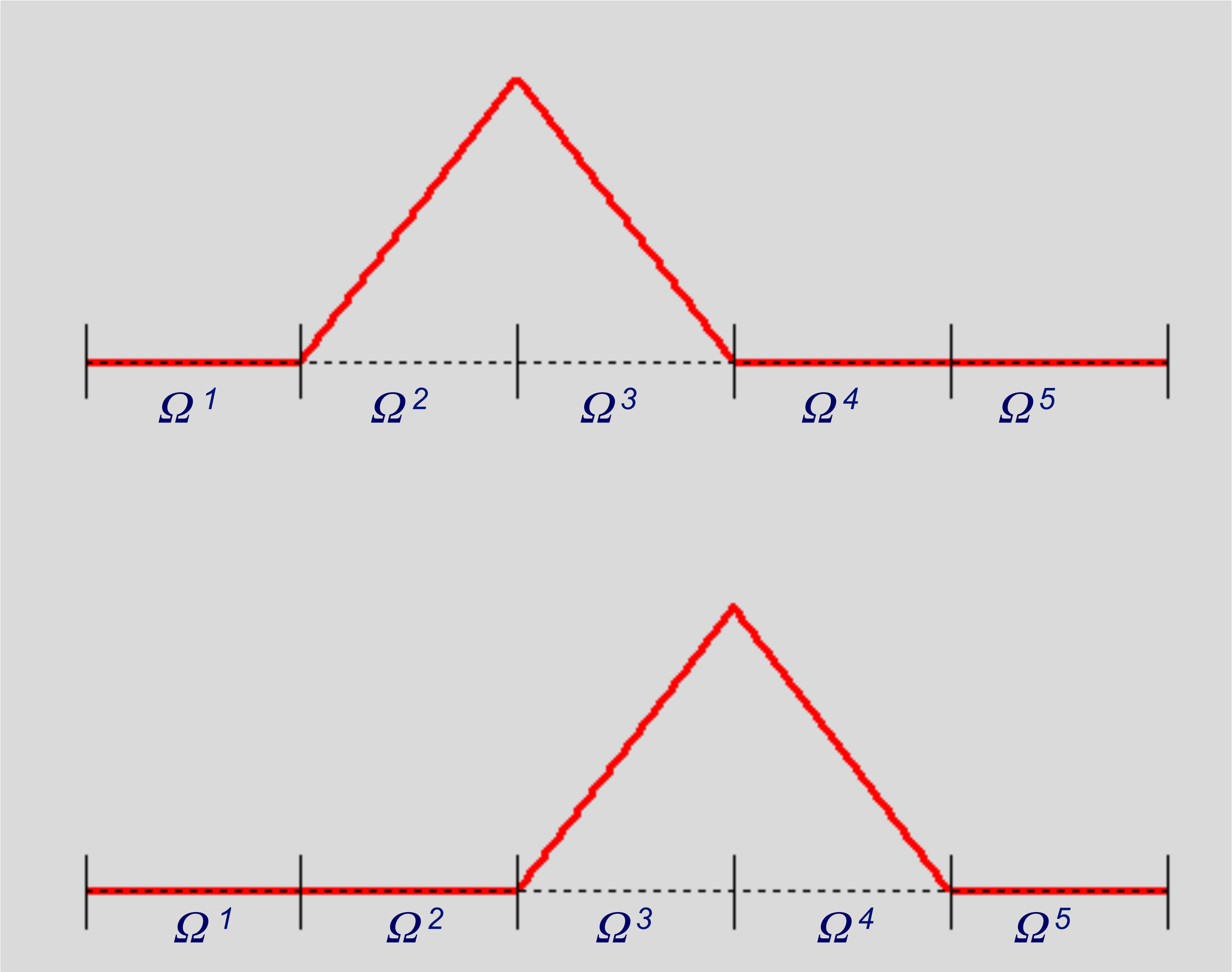
Figure 2: Examples of one-dimensional piecewise linear (left) and piecewise quadratic (right) Lagrangian basis functions, $\phi_2(x)$ and $\phi_3(x)$, with associated element support, $\Omega^e$, $e = 1, \ldots, E$.

- Linear case results in A being tridiagonal (b.w. = 1)
- Q: What is matrix bandwidth for piecewise quadratic case?

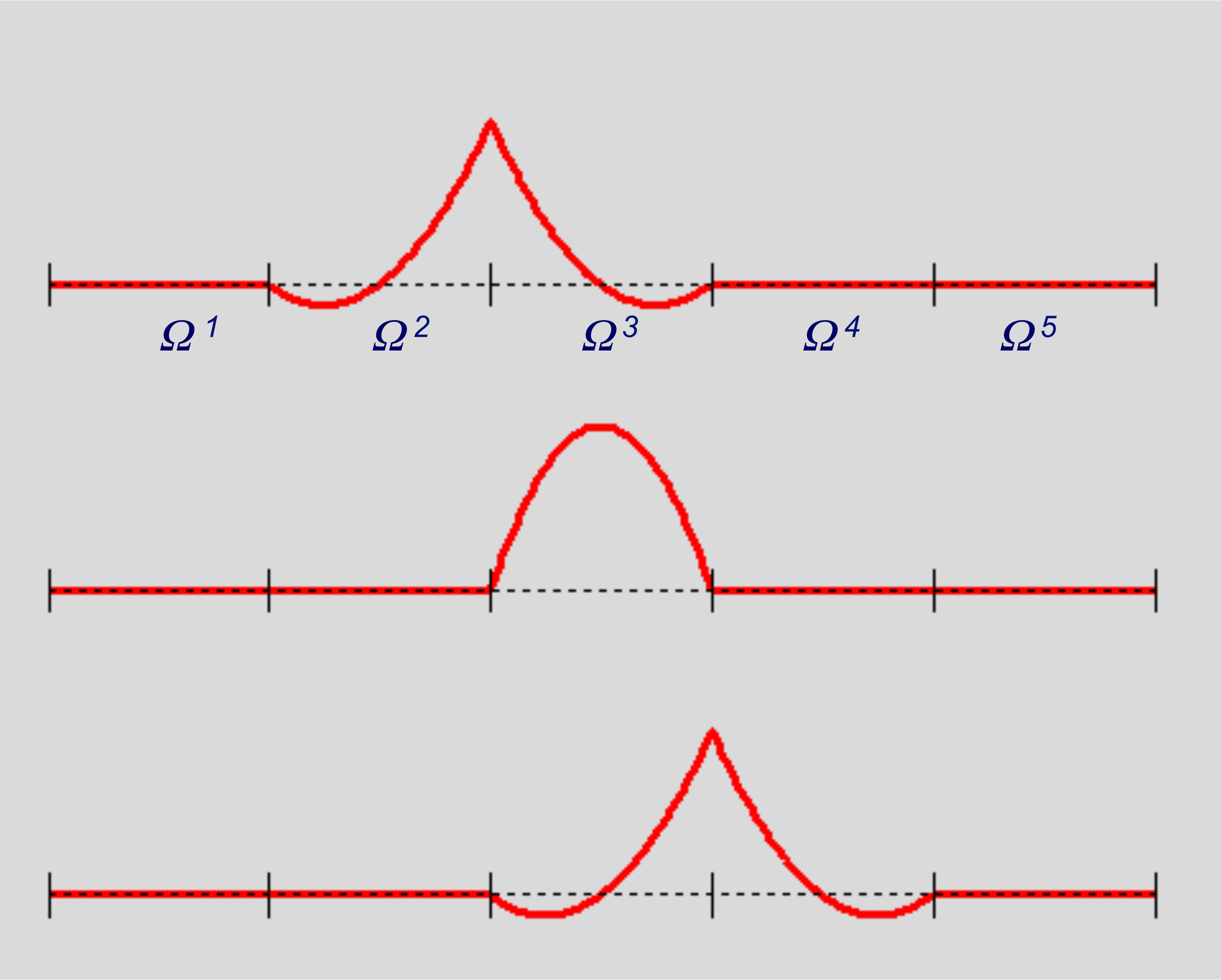
# Lagrange Polynomials: Good and Bad Point Distributions



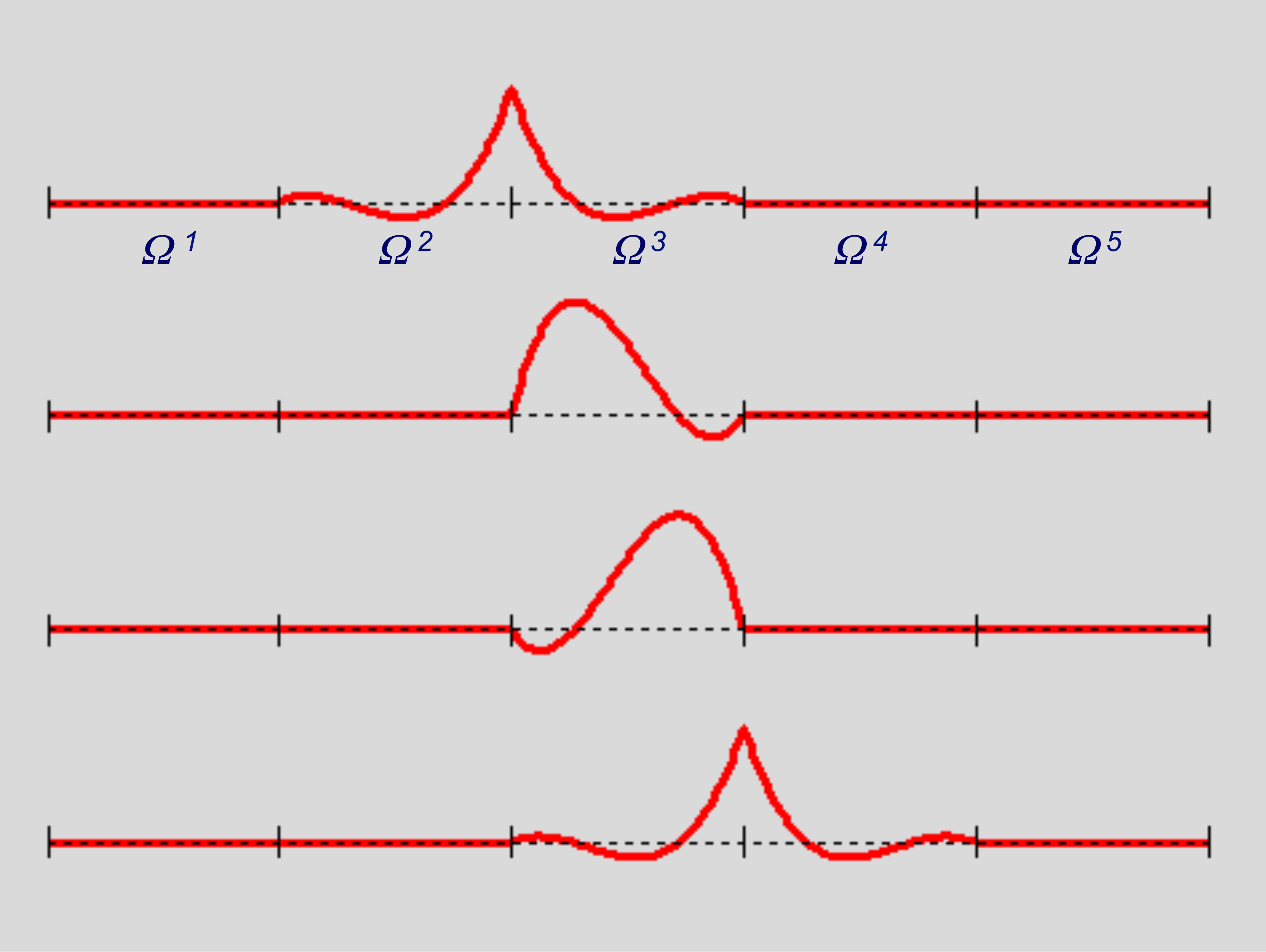
# Basis functions for N=1, E=5 on element 3.



# $\Omega^3$ basis functions for N=2, E=5



# $\Omega^3$ basis functions for N=3, E=5



# Important Properties of the Galerkin Formulation

- An essential property of the Galerkin formulation for the Poisson equation is that the solution is the ***best fit*** in the approximation space, with respect to the energy norm.

  Specifically, we consider the bilinear form,

$$a(v,u) \; := \; \int_0^1 \frac{dv}{dx}\frac{du}{dx}\,dx,$$

  and associated semi-norm,

$$||u||_a^2 \; := \; a(u,u),$$

  which is in fact a norm for all $u$ satisfying the boundary conditions.

- It is straightforward to show that our Galerkin solution, $u$, is the closest solution to the exact $\tilde{u}$ in the $a$-norm. That is,

$$\boxed{\| \, u - \tilde{u} \, \|_a \;\; \leq \;\; \| \, w - \tilde{u} \, \|_a \quad \textit{for all } w \in X_0^N}$$

- In fact, $u$ is closer to $\tilde{u}$ than the interpolant of $\tilde{u}$.

# Best Fit Property, 1/4

Define:

$$\begin{aligned}
\mathcal{L}^2_\Omega &= \left\{ v : \int_\Omega v^2 \, dx < \infty \right\} \\
\mathcal{H}^1 &= \left\{ v : v \in \mathcal{L}^2_\Omega, \int_\Omega (v')^2 \, dx < \infty \right\} \\
\mathcal{H}^1_0 &= \left\{ v : v \in \mathcal{H}^1, \; v|_{\partial\Omega} = 0 \right\}
\end{aligned}$$

Then, $\forall u, v \in \mathcal{H}^1_0$,

$$\begin{aligned}
a(u,v) &:= \int_\Omega u'v' \, dx && (a \text{ inner-product}) \\
||v||_a &:= \sqrt{a(v,v)} && (a\text{-norm}) \\
||\alpha v||_a &= |\alpha| \sqrt{a(v,v)} && \alpha \in \mathbb{R} \\
||v||_a &= 0 \text{ iff } v \equiv 0.
\end{aligned}$$

# Best Fit Property, 2/4

We now demonstrate that $||u - \tilde{u}||_a \leq ||w - \tilde{u}||_a \; \forall w \in X_0^N$.

Let $e := u - \tilde{u}$ and $v := w - u \in X_0^N$.

For any $w \in X_0^N$ we have

$$\begin{aligned}
||w - \tilde{u}||_a^2 &= ||v + u - \tilde{u}||_a^2 \\
&= ||v + e||_a^2 \\
&= \int_0^1 (v + e)' \, (v + e)' \, dx \\
&= \int_0^1 (v')^2 dx \; + \; 2 \int_0^1 v' \, e' dx \; + \; \int_0^1 (e')^2 dx
\end{aligned}$$

# Best Fit Property, 3/4

The second term vanishes:

$$
\begin{aligned}
\int_0^1 v'\, e' dx + \quad &= \int_0^1 v'\,(u - \tilde{u})' dx \\
&= \int_0^1 v'\, u'\, dx \; - \; \int_0^1 v'\, \tilde{u}'\, dx \\
&= \int_0^1 v'\, u'\, dx \; + \; \int_0^1 v\, \tilde{u}''\, dx \; - \; v\, \tilde{u}'\Big|_0^1 \\
&= \int_0^1 v'\, u'\, dx \; - \; \int_0^1 v\, f\, dx \\
&= 0 \qquad \forall\, v \in X_0^N,
\end{aligned}
$$

by construction of $u$.

# Best Fit Property 4/4

In summary, for any $w \in X_0^N$ we hawe

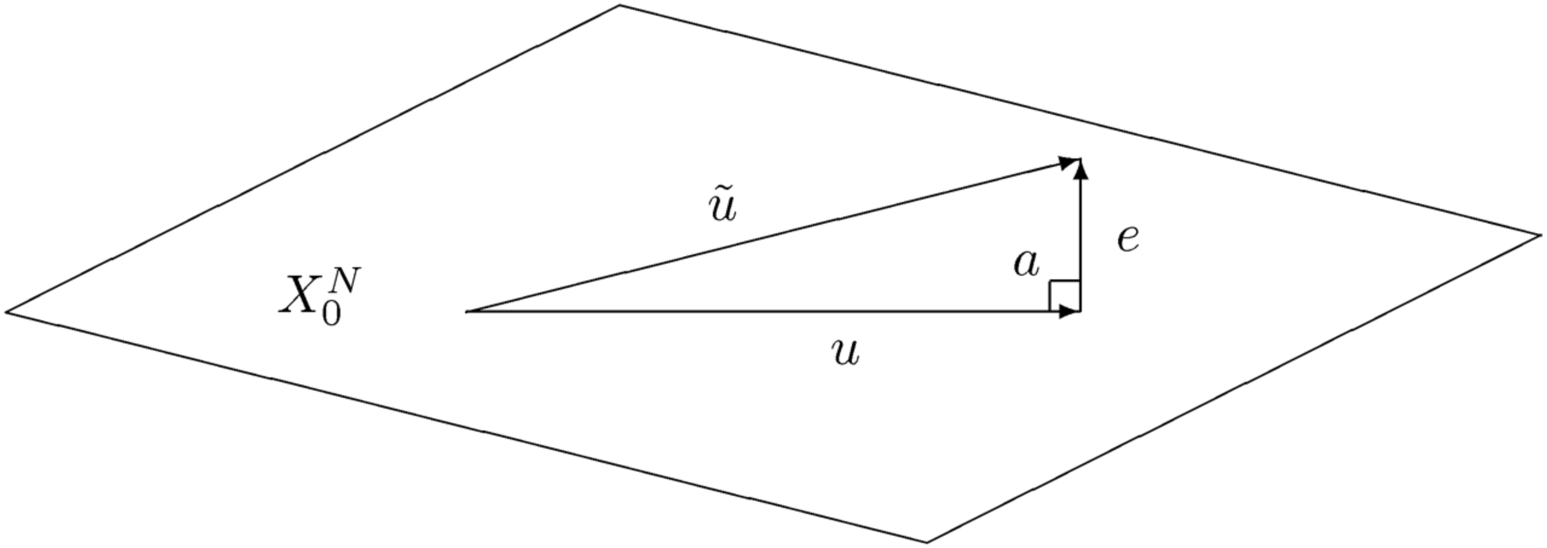
$$\begin{aligned}
||w - \tilde{u}||_a^2 &= ||v + u - \tilde{u}||_a^2 \\
&= ||v + e||_a^2 \\
&= \int_0^1 (v')^2 dx \; + \; 2 \int_0^1 \cancel{v' \, e' dx} \; + \; \int_0^1 (e')^2 dx \\
&= \int_0^1 (v')^2 dx \; + \; \int_0^1 (e')^2 dx \\
&\geq \int_0^1 (e')^2 dx \; = \; ||u - \tilde{u}||_a^2
\end{aligned}$$

Thus, of *all* functions in $X_0^N$, $u$ is the *closest* to $\tilde{u}$ in the $a$-norm.

A deeper analysis establishes that, for $\tilde{u}$ analytic and $X^N = \mathbb{P}_N$, one has

$$||\tilde{u} - u||_{\mathcal{H}_0^1} \; \leq \; C \, e^{-\gamma N}$$

# Best Fit Viewed as a Projection



- Note that this result also demonstrates that $a(v, e) = 0$ for all $v \in X_0^N$.

- That is, the Galerkin statement is equivalent to having the *error*,

$$e := u - \tilde{u} \perp_a X_0^N .$$

- Thus, $u$ is the *projection* of $\tilde{u}$ onto $X_0^N$ in the $a$ inner-product.

- The procedure is often referred to as a Galerkin projection.

# Best Fit Property Summary

- A significant advantage of the WRT over collocation is that the choice of basis for a given space, $X_0^N$, is immaterial – you will get the same answer for any basis, modulo round-off considerations, because of the ***best fit property***.

- That is, the choice of $\phi_i$ influences the condition number of A, but would give the same answer (in inifinite-precision arithmetic) whether one used Lagrange polynomials on uniform points, Chebyshev points, or even monomial bases.

- This is not the case for collocation – so WRT is much more robust.

- Of course, Lagrange polynomials on Chebyshev or Gauss-Lobatto-Legndre points are preferred from a conditioning standpoint.

# Eigenvalue Problems

- Standard eigenvalue problem for second-order ODE has form

$$u'' = \lambda f(t, u, u'), \qquad a < t < b$$

  with BC

$$u(a) = \alpha, \qquad u(b) = \beta$$

  where we seek not only solution $u$ but also parameter $\lambda$

- Scalar $\lambda$ (possibly complex) is *eigenvalue* and solution $u$ is corresponding *eigenfunction* for this two-point BVP

- Discretization of eigenvalue problem for ODE results in algebraic eigenvalue problem whose solution approximates that of original problem

# Example: Eigenvalue Problem

- Consider linear two-point BVP

$$u'' = \lambda g(t) u, \qquad a < t < b$$

with BC

$$u(a) = 0, \qquad u(b) = 0$$

- Introduce discrete mesh points $t_i$ in interval $[a, b]$, with mesh spacing $h$ and use standard finite difference approximation for second derivative to obtain algebraic system

$$\frac{y_{i+1} - 2y_i + y_{i-1}}{h^2} = \lambda g_i y_i, \quad i = 1, \dots, n$$

where $y_i = u(t_i)$ and $g_i = g(t_i)$, and from BC $y_0 = u(a) = 0$ and $y_{n+1} = u(b) = 0$

# Example, continued

- Assuming $g_i \neq 0$, divide equation $i$ by $g_i$ for $i = 1, \ldots, n$, to obtain linear system

$$\boldsymbol{A}\boldsymbol{y} = \lambda \boldsymbol{y}$$

where $n \times n$ matrix $\boldsymbol{A}$ has tridiagonal form

$$\boldsymbol{A} = \frac{1}{h^2} \begin{bmatrix} -2/g_1 & 1/g_1 & 0 & \cdots & 0 \\ 1/g_2 & -2/g_2 & 1/g_2 & \ddots & \vdots \\ 0 & \ddots & \ddots & \ddots & 0 \\ \vdots & \ddots & 1/g_{n-1} & -2/g_{n-1} & 1/g_{n-1} \\ 0 & \cdots & 0 & 1/g_n & -2/g_n \end{bmatrix}$$

- This standard algebraic eigenvalue problem can be solved by methods discussed previously