CS 450 – Numerical Analysis

Chapter 10: Boundary Value Problems for Ordinary Differential Equations †

Prof. Michael T. Heath

Department of Computer Science
University of Illinois at Urbana-Champaign
heath@illinois.edu

January 28, 2019

---

† Lecture slides based on the textbook *Scientific Computing: An Introductory Survey* by Michael T. Heath, copyright © 2018 by the Society for Industrial and Applied Mathematics. http://www.siam.org/books/cl80

Boundary Value Problems

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
- kth 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.

Boundary Value Problems, continued

- General *first-order two-point BVP* has form

  $$\boldsymbol{y}' = \boldsymbol{f}(t, \boldsymbol{y}), \qquad a < t < b$$

  with BC

  $$\boldsymbol{g}(\boldsymbol{y}(a), \boldsymbol{y}(b)) = \boldsymbol{0}$$

  where $\boldsymbol{f} \colon \mathbb{R}^{n+1} \to \mathbb{R}^n$ and $\boldsymbol{g} \colon \mathbb{R}^{2n} \to \mathbb{R}^n$

- Boundary conditions are *separated* if any given component of $\boldsymbol{g}$ involves solution values only at a or at b, but not both

- Boundary conditions are *linear* if they are of form

  $$\boldsymbol{B}_a \, \boldsymbol{y}(a) + \boldsymbol{B}_b \, \boldsymbol{y}(b) = \boldsymbol{c}$$

  where $\boldsymbol{B}_a, \boldsymbol{B}_b \in \mathbb{R}^{n \times n}$ and $\boldsymbol{c} \in \mathbb{R}^n$

- BVP is *linear* if ODE and BC are both linear

Example: Separated Linear Boundary Conditions

- Two-point BVP for second-order scalar ODE

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

with BC

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

is equivalent to first-order system of ODEs

$$\begin{bmatrix} y_1' \\ y_2' \end{bmatrix} = \begin{bmatrix} y_2 \\ f(t, y_1, y_2) \end{bmatrix}, \qquad a < t < b$$

with separated linear BC

$$\begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} y_1(a) \\ y_2(a) \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} y_1(b) \\ y_2(b) \end{bmatrix} = \begin{bmatrix} \alpha \\ \beta \end{bmatrix}$$

Existence and Uniqueness

- Unlike IVP, with BVP we cannot begin at initial point and continue solution step by step to nearby points

- Instead, solution is determined everywhere simultaneously, so existence and/or uniqueness may not hold

- For example,

  $$u'' = -u, \qquad 0 < t < b$$

  with BC

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

  with b integer multiple of π, has infinitely many solutions if $\beta = 0$, but no solution if $\beta \neq 0$

Existence and Uniqueness, continued

- In general, solvability of BVP

$$\boldsymbol{y}' = \boldsymbol{f}(t, \boldsymbol{y}), \qquad a < t < b$$

with BC

$$\boldsymbol{g}(\boldsymbol{y}(a), \boldsymbol{y}(b)) = \mathbf{0}$$

depends on solvability of algebraic equation

$$\boldsymbol{g}(\boldsymbol{x}, \boldsymbol{y}(b; \boldsymbol{x})) = \mathbf{0}$$

where $\boldsymbol{y}(t; \boldsymbol{x})$ denotes solution to ODE with initial condition $\boldsymbol{y}(a) = \boldsymbol{x}$ for $\boldsymbol{x} \in \mathbb{R}^n$

- Solvability of latter system is difficult to establish if $\boldsymbol{g}$ is nonlinear

Existence and Uniqueness, continued

- For *linear* BVP, existence and uniqueness are more tractable
- Consider linear BVP

  $$\boldsymbol{y}' = \boldsymbol{A}(t)\,\boldsymbol{y} + \boldsymbol{b}(t), \qquad a < t < b$$

  where $\boldsymbol{A}(t)$ and $\boldsymbol{b}(t)$ are continuous, with BC

  $$\boldsymbol{B}_a\,\boldsymbol{y}(a) + \boldsymbol{B}_b\,\boldsymbol{y}(b) = \boldsymbol{c}$$

- Let $\boldsymbol{Y}(t)$ denote matrix whose ith column, $\boldsymbol{y}_i(t)$, called ith *mode*, is solution to $\boldsymbol{y}' = \boldsymbol{A}(t)\boldsymbol{y}$ with initial condition $\boldsymbol{y}(a) = \boldsymbol{e}_i$, ith column of identity matrix
- Then BVP has unique solution if, and only if, matrix

  $$\boldsymbol{Q} \equiv \boldsymbol{B}_a\boldsymbol{Y}(a) + \boldsymbol{B}_b\boldsymbol{Y}(b)$$

  is nonsingular

Existence and Uniqueness, continued

- Assuming $\boldsymbol{Q}$ is nonsingular, define

$$\boldsymbol{\Phi}(t) = \boldsymbol{Y}(t)\,\boldsymbol{Q}^{-1}$$

and *Green's function*

$$\boldsymbol{G}(t,s) = \begin{cases} \boldsymbol{\Phi}(t)\boldsymbol{B}_a\boldsymbol{\Phi}(a)\boldsymbol{\Phi}^{-1}(s), & a \le s \le t \\ -\boldsymbol{\Phi}(t)\boldsymbol{B}_b\,\boldsymbol{\Phi}(b)\boldsymbol{\Phi}^{-1}(s), & t < s \le b \end{cases}$$

- Then solution to BVP given by

$$\boldsymbol{y}(t) = \boldsymbol{\Phi}(t)\,\boldsymbol{c} + \int_a^b \boldsymbol{G}(t,s)\,\boldsymbol{b}(s)\,ds$$

- This result also gives absolute condition number for BVP

$$\kappa = \max\{\|\boldsymbol{\Phi}\|_\infty,\ \|\boldsymbol{G}\|_\infty\}$$

Conditioning and Stability

- Conditioning or stability of BVP depends on interplay between growth of solution modes and boundary conditions
- For IVP, instability is associated with modes that grow exponentially as time increases
- For BVP, solution is determined everywhere simultaneously, so there is no notion of "direction" of integration in interval $[a, b]$
- Growth of modes increasing with time is limited by boundary conditions at b, and "growth" (in reverse) of decaying modes is limited by boundary conditions at a
- For BVP to be well-conditioned, growing and decaying modes must be controlled appropriately by boundary conditions imposed

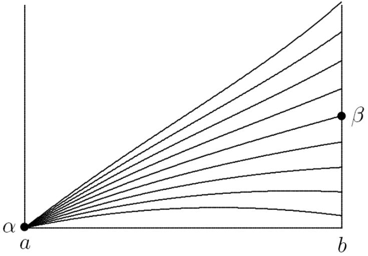
Numerical Methods for BVPs

Numerical Methods for BVPs

- For IVP, initial data supply all information necessary to begin numerical solution method at initial point and step forward from there
- For BVP, we have insufficient information to begin step-by-step numerical method, so numerical methods for solving BVPs are more complicated than those for solving IVPs
- We will consider four types of numerical methods for two-point BVPs
  - Shooting
  - Finite difference
  - Collocation
  - Galerkin

Shooting Method

- In statement of two-point BVP, we are given value of $u(a)$
- If we also knew value of $u'(a)$, then we would have IVP that we could solve by methods discussed previously
- Lacking that information, we try sequence of increasingly accurate guesses until we find value for $u'(a)$ such that when we solve resulting IVP, approximate solution value at $t = b$ matches desired boundary value, $u(b) = \beta$



Shooting Method, continued

- For given γ, value at b of solution $u(b)$ to IVP

$$u'' = f(t, u, u')$$

  with initial conditions

$$u(a) = \alpha, \qquad u'(a) = \gamma$$

  can be considered as function of γ, say $g(\gamma)$

- Then BVP becomes problem of solving equation $g(\gamma) = \beta$

- One-dimensional zero finder can be used to solve this scalar equation

Example: Shooting Method

- Consider two-point BVP for second-order ODE

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

with BC

$$u(0) = 0, \qquad u(1) = 1$$

- For each guess for $u'(0)$, we will integrate resulting IVP using classical fourth-order Runge-Kutta method to determine how close we come to hitting desired solution value at $t = 1$

- For simplicity of illustration, we will use step size $h = 0.5$ to integrate IVP from $t = 0$ to $t = 1$ in only two steps

- First, we transform second-order ODE into system of two first-order ODEs

$$\boldsymbol{y}'(t) = \begin{bmatrix} y_1'(t) \\ y_2'(t) \end{bmatrix} = \begin{bmatrix} y_2 \\ 6t \end{bmatrix}$$

Example, continued

- We first try guess for initial slope of $y_2(0) = 1$

$$
\begin{aligned}
\boldsymbol{y}^{(1)} &= \boldsymbol{y}^{(0)} + \frac{h}{6}(k_1 + 2k_2 + 2k_3 + k_4) \\
&= \begin{bmatrix} 0 \\ 1 \end{bmatrix} + \frac{0.5}{6}\left(\begin{bmatrix} 1 \\ 0 \end{bmatrix} + 2\begin{bmatrix} 1.0 \\ 1.5 \end{bmatrix} + 2\begin{bmatrix} 1.375 \\ 1.500 \end{bmatrix} + \begin{bmatrix} 1.75 \\ 3.00 \end{bmatrix} \right) = \begin{bmatrix} 0.625 \\ 1.750 \end{bmatrix} \\
\boldsymbol{y}^{(2)} &= \begin{bmatrix} 0.625 \\ 1.750 \end{bmatrix} + \frac{0.5}{6}\left(\begin{bmatrix} 1.75 \\ 3.00 \end{bmatrix} + 2\begin{bmatrix} 2.5 \\ 4.5 \end{bmatrix} + 2\begin{bmatrix} 2.875 \\ 4.500 \end{bmatrix} + \begin{bmatrix} 4 \\ 6 \end{bmatrix} \right) = \begin{bmatrix} 2 \\ 4 \end{bmatrix}
\end{aligned}
$$

- So we have hit $y_1(1) = 2$ instead of desired value $y_1(1) = 1$

Example, continued

- We try again, this time with initial slope $y_2(0) = -1$

$$
\begin{aligned}
\boldsymbol{y}^{(1)} &= \begin{bmatrix} 0 \\ -1 \end{bmatrix} + \frac{0.5}{6}\left(\begin{bmatrix} -1 \\ 0 \end{bmatrix} + 2\begin{bmatrix} -1.0 \\ 1.5 \end{bmatrix} + 2\begin{bmatrix} -0.625 \\ 1.500 \end{bmatrix} + \begin{bmatrix} -0.25 \\ 3.00 \end{bmatrix} \right) \\
&= \begin{bmatrix} -0.375 \\ -0.250 \end{bmatrix} \\
\boldsymbol{y}^{(2)} &= \begin{bmatrix} -0.375 \\ -0.250 \end{bmatrix} + \frac{0.5}{6}\left(\begin{bmatrix} -0.25 \\ 3.00 \end{bmatrix} + 2\begin{bmatrix} 0.5 \\ 4.5 \end{bmatrix} + 2\begin{bmatrix} 0.875 \\ 4.500 \end{bmatrix} + \begin{bmatrix} 2 \\ 6 \end{bmatrix} \right) \\
&= \begin{bmatrix} 0 \\ 2 \end{bmatrix}
\end{aligned}
$$

- So we have hit $y_1(1) = 0$ instead of desired value $y_1(1) = 1$, but we now have initial slope bracketed between -1 and 1

Example, continued

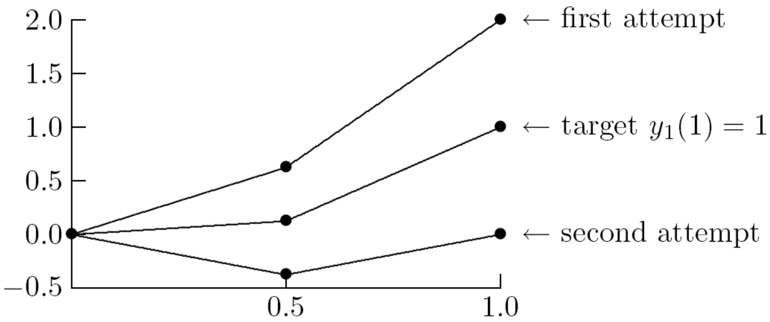
- We omit further iterations necessary to identify correct initial slope, which turns out to be $y_2(0) = 0$

$$\begin{aligned}\boldsymbol{y}^{(1)} &= \begin{bmatrix}0\\0\end{bmatrix} + \frac{0.5}{6}\left(\begin{bmatrix}0\\0\end{bmatrix} + 2\begin{bmatrix}0.0\\1.5\end{bmatrix} + 2\begin{bmatrix}0.375\\1.500\end{bmatrix} + \begin{bmatrix}0.75\\3.00\end{bmatrix}\right)\\ &= \begin{bmatrix}0.125\\0.750\end{bmatrix}\end{aligned}$$

$$\begin{aligned}\boldsymbol{y}^{(2)} &= \begin{bmatrix}0.125\\0.750\end{bmatrix} + \frac{0.5}{6}\left(\begin{bmatrix}0.75\\3.00\end{bmatrix} + 2\begin{bmatrix}1.5\\4.5\end{bmatrix} + 2\begin{bmatrix}1.875\\4.500\end{bmatrix} + \begin{bmatrix}3\\6\end{bmatrix}\right)\\ &= \begin{bmatrix}1\\3\end{bmatrix}\end{aligned}$$

- So we have indeed hit target solution value $y_1(1) = 1$

Example, continued



⟨ interactive example ⟩

Multiple Shooting

- Simple shooting method inherits stability (or instability) of associated IVP, which may be unstable even when BVP is stable
- Such ill-conditioning may make it difficult to achieve convergence of iterative method for solving nonlinear equation
- Potential remedy is *multiple shooting*, in which interval $[a, b]$ is divided into subintervals, and shooting is carried out on each
- Requiring continuity at internal mesh points provides BC for individual subproblems
- Multiple shooting results in larger *system* of nonlinear equations to solve

Finite Difference Method

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

- 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: Finite Difference Method

- Consider again two-point BVP

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

with BC

$$u(0) = 0, \qquad u(1) = 1$$

- To keep computation to minimum, we compute approximate solution at one interior mesh point, $t = 0.5$, in interval $[0, 1]$
- Including boundary points, we have three mesh points, $t_0 = 0$, $t_1 = 0.5$, and $t_2 = 1$
- From BC, we know that $y_0 = u(t_0) = 0$ and $y_2 = u(t_2) = 1$, and we seek approximate solution $y_1 \approx u(t_1)$

Example, continued

- Replacing derivatives by standard finite difference approximations at t_1 gives equation

$$\frac{y_2 - 2y_1 + y_0}{h^2} = f\left(t_1, y_1, \frac{y_2 - y_0}{2h}\right)$$

- Substituting boundary data, mesh size, and right-hand side for this example we obtain

$$\frac{1 - 2y_1 + 0}{(0.5)^2} = 6t_1$$

or

$$4 - 8y_1 = 6(0.5) = 3$$

so that

$$y(0.5) \approx y_1 = 1/8 = 0.125$$

Example, continued

- In a practical problem, much smaller step size and many more mesh points would be required to achieve acceptable accuracy
- We would therefore obtain *system* of equations to solve for approximate solution values at mesh points, rather than single equation as in this example

⟨ interactive example ⟩

Collocation Method

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

Collocation Method

- Popular choices of basis functions include polynomials, B-splines, and trigonometric functions
- Basis functions with global support, such as polynomials or trigonometric functions, yield *spectral method*
- Basis functions with highly localized support, such as B-splines, yield *finite element method*

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 $\boldsymbol{x}$

Example: Collocation Method

- Consider again two-point BVP

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

with BC

$$u(0) = 0, \qquad u(1) = 1$$

- To keep computation to minimum, we use one interior collocation point, $t = 0.5$

- Including boundary points, we have three collocation points, $t_0 = 0$, $t_1 = 0.5$, and $t_2 = 1$, so we will be able to determine three parameters

- As basis functions we use first three monomials, so approximate solution has form

$$v(t, \boldsymbol{x}) = x_1 + x_2 t + x_3 t^2$$

Example, continued

- Derivatives of approximate solution function with respect to t are given by
$$v'(t, \boldsymbol{x}) = x_2 + 2x_3 t, \qquad v''(t, \boldsymbol{x}) = 2x_3$$
- Requiring ODE to be satisfied at interior collocation point $t_2 = 0.5$ gives equation
$$v''(t_2, \boldsymbol{x}) = f(t_2, v(t_2, \boldsymbol{x}), v'(t_2, \boldsymbol{x}))$$
or
$$2x_3 = 6t_2 = 6(0.5) = 3$$
- Boundary condition at $t_1 = 0$ gives equation
$$x_1 + x_2 t_1 + x_3 t_1^2 = x_1 = 0$$
- Boundary condition at $t_3 = 1$ gives equation
$$x_1 + x_2 t_3 + x_3 t_3^2 = x_1 + x_2 + x_3 = 1$$

Example, continued

- Solving this system of three equations in three unknowns gives

$$x_1 = 0, \qquad x_2 = -0.5, \qquad x_3 = 1.5$$

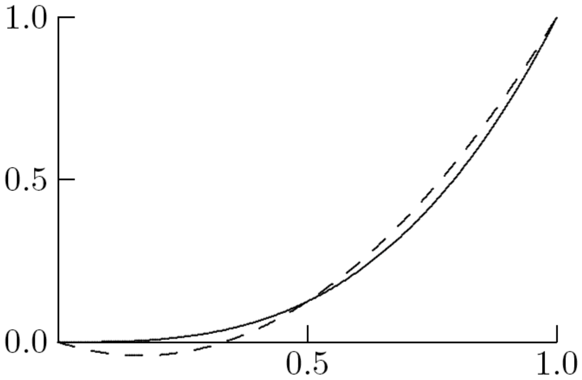
so approximate solution function is quadratic polynomial

$$u(t) \approx v(t, \boldsymbol{x}) = -0.5t + 1.5t^2$$

- At interior collocation point, $t_2 = 0.5$, we have approximate solution value

$$u(0.5) \approx v(0.5, \boldsymbol{x}) = 0.125$$

Example, continued



⟨ interactive example ⟩

Galerkin Method

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

- Using *least squares method*, we can minimize

  $$F(\boldsymbol{x}) = \tfrac{1}{2} \int_a^b r(t, \boldsymbol{x})^2 \, dt$$

  by setting each component of its gradient to zero

- This yields symmetric system of linear algebraic equations $\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}$, where

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

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

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,

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

- This yields linear system $\boldsymbol{Ax} = \boldsymbol{b}$, 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., $w_i = \phi_i$, $i = 1, \ldots, n$

- Orthogonality condition then becomes

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

  or

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

Galerkin Method, continued

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

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

- Assuming basis functions ϕ_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

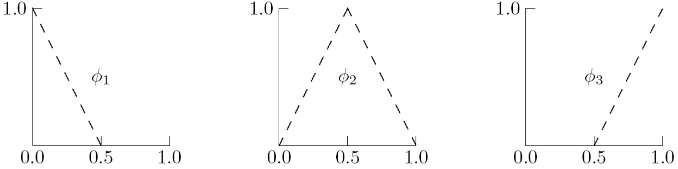
Example: Galerkin Method

- 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 ϕ_2 and obtain equation

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

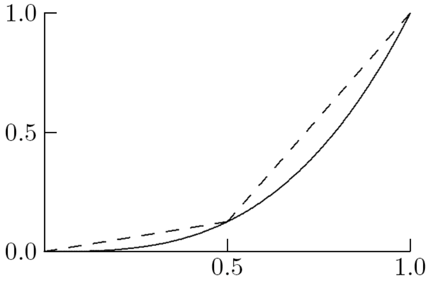
or, upon evaluating these simple integrals analytically

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

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 ⟩

Eigenvalue Problems

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 λ

- Scalar λ (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

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

  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 $i = 1, \dots, n$, 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

Summary – ODE Boundary Value Problems

- Two-point BVP for ODE specifies BC at both endpoints of interval
- *Shooting* method replaces BVP by sequence of IVPs, with missing initial conditions determined by nonlinear equation solver
- *Finite difference* method replaces derivatives in ODE by finite differences defined on mesh of points, resulting in system of linear algebraic equations to solve for sample values of ODE solution
- *Collocation* method approximates ODE solution by linear combination of suitably smooth basis functions, with coefficients determined by requiring approximate solution to satisfy ODE at discrete set of collocation points
- *Galerkin* method approximates ODE solution by linear combination of basis functions, with coefficients determined by requiring residual to be orthogonal to each basis function