CS 450: Numerical Analysis\(^1\)
Boundary Value Problems for Ordinary Differential Equations

University of Illinois at Urbana-Champaign

\(^1\)These slides have been drafted by Edgar Solomonik as lecture templates and supplementary material for the book “Scientific Computing: An Introductory Survey” by Michael T. Heath (slides).
Boundary Conditions

- Often we seek to solve a differential equation that satisfies conditions on its values and derivatives on parts of the domain boundary.
  - *Dirichlet boundary conditions* specify values of \( y(t) \) at boundary.
  - *Neumann boundary conditions* specify values of derivative \( f(t, y) \) at boundary.

- Consider a first order ODE \( y'(t) = f(t, y) \) with *linear boundary conditions* on domain \( t \in [a, b] \):
  \[
  B_a y(a) + B_b y(b) = c
  \]

- IVPs are a special case of Dirichlet condition with \( B_a = I \), \( B_b = 0 \).
- Conditions are *separated* if they do not couple different boundary points, i.e., for all \( i \), the \( i \)th row of either \( B_a \) or \( B_b \) is zero.
- Higher-order boundary conditions can be reduced to linear boundary conditions in the same way as a nonlinear ODE is reduced to a linear ODE.
Existence of Solutions for Linear ODE BVPs

- The solutions of linear ODE BVP \( y'(t) = A(t)y(t) + b(t) \) are linear combinations of solutions to linear homogeneous ODE IVPs \( y'(t) = A(t)y(t) \):

  - Let the solutions \( y_i(t) \) to the homogeneous ODE, \( y_i'(t) = A(t)y_i(t) \), with initial conditions \( y_i(a) = e_i \) be columns of
    \[
    Y(t) = \begin{bmatrix} y_1(t) & \cdots & y_n(t) \end{bmatrix} = I + \int_a^t A(s)Y(s)ds.
    \]

  - The ODE BVP solutions are then given by \( y(t) = Y(t)u(t) \) for some \( u(t) \), with
    \[
    y'(t) = A(t)y(t) + b(t) \quad \Rightarrow \quad Y'(t)u(t) + Y(t)u'(t) = A(t)Y(t)u(t) + b(t),
    \]
    \[
    Y'(t) = A(t)Y(t) \quad \Rightarrow \quad u'(t) = Y(t)^{-1}b(t).
    \]

- Solution \( u(t) \) (and \( y(t) \)) exists if \( Q = B_aY(a) + B_bY(b) \) is invertible:

  \[
  B_aY(a)u(a) + B_bY(b)\left(u(a) + \int_a^b u'(s)ds\right) = c,
  \]
  \[
  u(a) = \left(Q\begin{bmatrix} B_aY(a) + B_bY(b) \end{bmatrix}^{-1}\left(c - B_bY(b) \int_a^b u'(s)ds\right)\right).
  \]
Green’s Function Form of Solution for Linear ODE BVPs

For any given $b(t)$ and $c$, the solution to the BVP can be written in the form:

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

$\Phi(t) = Y(t)Q^{-1}$ is the fundamental matrix and the Green’s function is

$$G(t, s) = Y(t)Q^{-1}I(s)Y^{-1}(s), \quad I(s) = \begin{cases} B_aY(a) & : s < t \\ -B_bY(b) & : s \geq t \end{cases}$$

From our expression for $u(a)$ and the integral equation for $y(t)$,

$$y(t) = Y(t)Q^{-1}\left(c - B_bY(b)\int_a^b u'(s)ds\right) + Y(t)\int_a^t u'(s)ds$$

$$= \Phi(t)c + Y(t)Q^{-1}\left(- B_bY(b)\int_a^b u'(s)ds + Q\int_a^t u'(s)ds\right)$$

$$= \Phi(t)c + Y(t)Q^{-1}\left( B_aY(a)\int_a^t Y^{-1}(s)b(s)ds - B_bY(b)\int_t^b Y^{-1}(s)b(s)\right).$$
Conditioning of Linear ODE BVPs

- For any given \( b(t) \) and \( c \), the solution to the BVP can be written in the form:

\[
y(t) = \Phi(t)c + \int_{a}^{b} G(t,s)b(s)ds
\]

\( \Phi(t) = Y(t)Q^{-1} \) is the fundamental matrix, which like the Green's function is associated with the homogeneous ODE as well as its linear boundary condition matrices \( B_a \) and \( B_b \), but is independent \( b(t) \) and \( c \).

- The absolute condition number of the BVP is \( \kappa = \max\{||\Phi||_{\infty}, ||G||_{\infty}\} \):

This sensitivity measure enables us to bound the perturbation \( ||\hat{y} - y||_{\infty} \) with respect to the magnitude of a perturbation to \( b(t) \) or \( c \).
Shooting Method for ODE BVPs

For linear ODEs, we construct solutions from IVP solutions in \( Y(t) \), which suggests the **shooting method** for solving BVPs by reduction to IVPs:

*For \( k = 1, 2, \ldots \) repeat until convergence:*

1. construct approximate initial value guesses \( \hat{y}^{(k)}(a) \approx y(a) \),
2. solve the resulting IVP,
3. check the quality of the solution at the new boundary,
   \[ \| B_b \hat{y}^{(k)}(b) - B_a \hat{y}^{(k)}(a) - c \|, \]
4. pick the initial conditions for the next shot, \( \hat{y}^{(k+1)}(a) \) by treating \( \hat{y}^{(l)}(a) \) for \( l = 1, \ldots, k \) as guesses \( x^{(1)}, \ldots, x^{(k)} \) to root finding procedure for
   \[ h(x) = B_a x + B_b y_x(b) - c, \text{ where } y_x(b) \text{ is the IVP solution with } y_x(a) = x. \]

**Multiple shooting** employs the shooting method over subdomains:

- The shooting problems on subdomains are interdependent, as they must satisfy continuity conditions on boundaries between them, leading to a system of nonlinear equations.
- Improves on conditioning of shooting method, which can suffer from ill-conditioning of large IVPs.
Finite Difference Methods

Rather than solve a sequence of IVPs that satisfy the ODEs until they (approximately) satisfy boundary conditions, we can refine an approximation that satisfies the boundary conditions, until it satisfies the ODE:

> Finite difference methods work by obtaining a solution on points \( t_1, \ldots, t_n \), so that \( \hat{y}_k \approx y(t_k) \) by finite-difference formulae, for example,

\[
f(t, y) = y'(t) \approx \frac{y(t + h) - y(t - h)}{2h} \Rightarrow f(t_k, \hat{y}_k) = \frac{\hat{y}_{k+1} - \hat{y}_{k-1}}{t_{k+1} - t_{k-1}}.
\]

> The resulting system of equations can be solved by standard methods and is linear if \( f \) is linear.

Convergence to solution is obtained with decreasing step size \( h \) so long as the method is consistent and stable:

> Consistency implies that the truncation error goes to zero.

> Stability ensures input perturbations have bounded effect on solution.
Let's derive the finite difference method for the ODE BVP defined by

\[ u'' + 7(1 + t^2)u = 0 \]

with boundary conditions \( u(-1) = 3 \) and \( u(1) = -3 \), using a centered difference approximation for \( u'' \) on \( t_1, \ldots, t_n, t_{i+1} - t_i = h \).

We have equations \( u(-1) = u(t_1) = u_1 = 3, \ u(1) = u(t_n) = u_n = 3 \) and \( n - 2 \) finite difference equations, one for each \( i \in \{2, \ldots, n - 1\} \),

\[ \frac{u_{i+1} - 2u_i + u_{i-1}}{h^2} + 7(1 + t_i^2)u_i = 0. \]

These correspond to a linear system based on matrices:

\[
A = \begin{bmatrix}
1 & -2/h^2 & 1/h^2 \\
1/h^2 & \ddots & \ddots & 1/h^2 \\
& \ddots & \ddots & \ddots \\
1/h^2 & -2/h^2 & 1/h^2 & 1
\end{bmatrix}
\quad \text{and} \quad
B = \begin{bmatrix}
0 & & & & 7(1 + t_2^2) \\
0 & \ddots & \ddots & \ddots & \ddots \\
& \ddots & \ddots & \ddots & \ddots \\
7(1 + t_{n-1}^2) & 0 & \ddots & \ddots & \ddots \\
0 & \ddots & \ddots & \ddots & \ddots
\end{bmatrix},
\]

where \((A + B)u = [3 \ 0 \ \cdots 0 \ -3]^T\).
Collocation Methods

- **Collocation methods** approximate $y$ by representing it in a basis

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

- Seek to satisfy for collocation points $t_1, \ldots, t_n$ with $t_1 = a$ and $t_n = b$,

$$\forall i \in \{2, \ldots, n-1\} \quad v'(t_i, x) = f(t_i, v(t_i, x)).$$

- Two more equations typically obtained from boundary conditions at $t_1, t_n$.

- Choices of basis functions give different families of methods:
  - **Spectral methods** use polynomials or trigonometric functions for $\phi_i$, which are nonzero over most of $[a, b]$, and have the advantage of corresponding to eigenfunctions of differential operators.
  - **Finite element** methods leverage basis functions with local support (e.g. B-splines) and yield sparsity in the resulting problem since many pairs of basis functions have disjoint support.

**Demo:** Sparse matrices
Solving BVPs by Optimization

- To improve robustness, define and minimize a residual error over the whole domain rather than at collocation points.
  - For simplified scenario $f(t, y) = f(t)$,
    \[
    r(t, x) = v'(t, x) - f(t) = \sum_{j=1}^{n} x_j \phi'_j(t) - f(t).
    \]
  - In particular, we seek to minimize the objective function,
    \[
    F(x) = \frac{1}{2} \int_{a}^{b} ||r(t, x)||^2 dt.
    \]
  - The first-order optimality conditions of the optimization problem are a system of linear equations $Ax = b$:
    \[
    0 = \frac{dF}{dx_i} = \int_{a}^{b} (r(t, x))^T \frac{dr}{dx_i} dt = \int_{a}^{b} (r(t, x))^T \phi'_i(t) dt
    = \sum_{j=1}^{n} x_j \int_{a}^{b} \phi'_j(t)^T \phi'_i(t) dt - \int_{a}^{b} f(t)^T \phi'_i(t) dt
    \]

Weighted Residual

- **Weighted residual methods** work by ensuring the residual is orthogonal with respect to a given set of weight functions:

  - Rather than setting components of the gradient to zero, we instead have
    \[
    \int_a^b r(t, x)^T w_i(t) \, dt = 0, \forall i \in \{1, \ldots, n\}.
    \]

  - Again, we obtain a system of equations of the form \( Ax = b \), where
    \[
    a_{ij} = \int_a^b \phi'_j(t)^T w_i(t), \quad b_i = \int_a^b f(t)^T w_i(t).
    \]

  - The collocation method is a weighted residual method where \( w_i(t) = \delta(t - t_i) \).

  - The **Galerkin method** is a weighted residual method where \( w_i = \phi_i \).

Linear system with the **stiffness matrix** \( A \) and **load vector** \( b \) is

\[
0 = \sum_{j=1}^n x_j \left[ \int_a^b \phi'_j(t)^T \phi_i(t) \, dt - \int_a^b f(t)^T \phi_i(t) \, dt \right].
\]
Second-Order BVPs: Poisson Equation

In practice, BVPs are at least second order and it's advantageous to work in the natural set of variables.

- Consider the **Poisson equation** \( u''(t) = f(t) \) with boundary conditions \( u(a) = u(b) = 0 \) and define a localized basis of hat functions:

\[
\phi_i(t) = \begin{cases} 
(t - t_{i-1})/h & : t \in [t_{i-1}, t_i] \\
(t_{i+1} - t)/h & : t \in [t_i, t_{i+1}] \\
0 & : \text{otherwise}
\end{cases}
\]

for \( i \in \{1, \ldots, n\} \), handling boundaries via \( t_0 = t_1 = a \) and \( t_n = t_{n+1} = b \).

- Defining residual equation by analogy to the first order case, we obtain,

\[
r = v'' - f, \quad \text{so that} \quad r(t, x) = \sum_{j=1}^{n} x_j \phi_j''(t) - f(t).
\]

However, with our choice of basis, \( \phi_j''(t) \) is undefined, since \( \phi_j'(t) \) is discontinuous at \( t_{j-1}, t_j, t_{j+1} \).
The finite-element method permits a lesser degree of differentiability of basis functions by casting the ODE in *weak form*:

For any solution \( u \), if test function \( \phi_i \) satisfies the boundary conditions, the ODE satisfies the weak form,

\[
\int_a^b f(t) \phi_i(t) dt = \int_a^b u''(t) \phi_i(t) dt = u'(b) \phi_i(b) - u'(a) \phi_i(a) - \int_a^b u'(t) \phi_i'(t) dt
\]

\[
= - \int_a^b u'(t) \phi_i'(t) dt.
\]

Note that the final equation contains no second derivatives, and subsequently we can form the linear system \( Ax = b \) with

\[
a_{ij} = - \int_a^b \phi_j'(t) \phi_i'(t) dt, \quad b_i = \int_a^b f(t) \phi_i(t) dt.
\]

The finite element method thus searches the larger (once-differentiable) function space to find a solution \( u \) that is in a (twice-differentiable) subspace.
A typical second-order scalar ODE BVP eigenvalue problem is to find eigenvalue $\lambda$ and eigenfunction $u$ to satisfy

$$ u'' = \lambda f(t, u, u'), \quad \text{with boundary conditions } u(a) = 0, u(b) = 0. $$

These can be solved, e.g. for $f(t, u, u') = g(t)u$ by finite differences:

- **Approximating the solution at a set of points $t_1, \ldots, t_n$ using finite differences,**

  $$ \frac{y_{i+1} - 2y_i + y_{i-1}}{h^2} = \lambda g_i y_i. $$

- **This yields a tridiagonal matrix eigenvalue problem $Ay = \lambda y$ where**

  $$ \frac{y_{i+1} - 2y_i + y_{i-1}}{g_i h^2} = \lambda y_i. $$
Generalized matrix eigenvalue problems arise from more sophisticated ODEs, 

\[ u'' = \lambda (g(t)u + h(t)u'), \quad \text{with boundary conditions } u(a) = 0, u(b) = 0. \]

Again approximate each of the derivatives at a set of points \( t_1, \ldots, t_n \) using finite differences,

\[ \frac{y_{i+1} - 2y_i + y_{i-1}}{h^2} = \lambda \left( g_i + \frac{y_{i+1} - y_{i-1}}{2h} \right) y_i. \]

These corresponds to a generalized matrix eigenvalue problem

\[ Ay = \lambda By, \]

where both \( A \) and \( B \) are tridiagonal.

Specialized methods exist for solving generalized matrix eigenvalue problems (also referred to as matrix pencil eigenvalue problems).