MS554 Lecture Notes: Boundary Value Problems (ODEs)
Lecture 12; Oct. 18, 2001

Objectives:

- Linear ordinary boundary value problems
- Finite differences
- Shooting method
- **Materials:** Gerald and Wheatley, Chapter 7; Golub and Ortega, Chapter 3; Hornberger and Wiberg; Chapter 5.

Ordinary, linear, differential equations

The general problem,

$$\frac{d^2}{dx^2}v(x) \;=\; b(x)\frac{d}{dx}v(x) + c(x)v(x) + d(x);\; 0 \leq x \leq 1 \tag{1}$$

is a second-order, ordinary differential equation- because it is a function only of x. It is linear if the functions $b(x)$, $c(x)$, and $d(x)$ depend only on x, and do not include $v(x)$. If v, b, c, or d were a function of more than one variable, it would be a partial differential equation instead of an o.d.e. To solve a second order linear differential equation, the problem statement (equ. 1) must be accompanied by two boundary values, such as $v(x = 0) \;=\; \alpha$, and $v(x = 1) \;=\; \beta$, and is then usually called a boundary value problem. We'll discuss two methods for solving linear boundary value problems; the shooting method and finite differences.

Types of boundary conditions

Four types of boundary conditions can usually be specified.

- **Dirichlet** boundary conditions specify the value of v(x) at the boundaries; as in the previous example ($v(x = 0) \;=\; \alpha$, and $v(x = 1) \;=\; \beta$).
- **Neumann** boundary conditions specify the gradient of the function v at a boundaries; for example $d[v(x = 0)]/dx \;=\; \gamma$.
- **Mixed** boundary conditions are a linear combination of Dirichlet and Neumann boundary conditions; such as $v(x = 1) + d[v(x = 1)]/dx \;=\; \delta$.
- **Periodic** boundary conditions state that the function values are equal at the two ends of the model domain; $v(0) = v(1)$.

For a 1^{st} order ODE, one boundary condition is needed; for a 2^{nd} order (as in equ. 1), two boundary conditions are needed; for a 3^{rd} order, three boundary conditions are needed, etc.

Finite differences

To solve the boundary value problem using finite differences, we use the discretizations of v'(x), v"(x), etc. that we derived in our section on numerical calculus:

$$\frac{dv}{dx} = \frac{v(x_{i+1}) - v(x_{i-1})}{x_{i+1} - x_{i-1}} + O(h^2) \tag{2}$$

$$\frac{d^2v}{dx^2} = \frac{v(x_{i+1}) - 2v(x_i) + v(x_{i-1})}{(x_{i+1} - x_i)(x_i - x_{i-1})} + O(h^2) \tag{3}$$

where $h = x_{i+1} - x_i$.

Example with Dirichlet boundary conditions

For example, for the general problem (equ. 1), suppose that $b(x) = 0$;

$$\begin{aligned} \frac{d^2v(x)}{dx^2} &= c(x)v(x) + d(x);\ for\ 0 \le x \le 1 \\ &\quad subject\ to\ v(0) = \alpha; v(1) = \beta; \end{aligned} \tag{4}$$

The first step is to break the interval $0 \le x \le 1$ into n intervals; $x_0 = 0$; $x_1 = h; \cdots, x_{n+1} = 1$; where $h = 1/(n+1)$. For each grid-point, $x_i,\ i = 1, \cdots n$, we can write

$$\frac{v_{i-1} - 2v_i + v_{i+1}}{h^2} = c_i v_i + d_i \tag{5}$$

This gives us n equations ($i = 1, \cdots n$), in n+2 unknowns ($v_0, v_1, \cdots, v_{n+1}$). They can be rewritten as

$$-v_{i-1} + \left(2 + h^2 c_i\right) v_i - v_{i+1} = h^2 d_i \tag{6}$$

If we then use our boundary conditions ($v_0 = \alpha$, $v_{n+1} = \beta$) we get the system of equations

$$\begin{aligned} \left(2 + h^2 c_1\right) v_1 - v_2 &= h^2 d_1 + \alpha; \\ -v_1 + \left(2 + h^2 c_2\right) v_2 - v_3 &= h^2 d_2; \\ -v_2 + \left(2 + h^2 c_3\right) v_3 - v_4 &= h^2 d_3; \\ &\vdots \\ -v_{n-2} + \left(2 + h^2 c_{n-1}\right) v_{n-1} - v_n &= h^2 d_{n-1}; \\ -v_{n-1} + \left(2 + h^2 c_n\right) v_n &= h^2 d_n + \beta; \end{aligned} \tag{7}$$

In matrix form this is $Av = d$:

```
| a_11 -1     0    . . .   0         0      0   | | v_1   | = | h^2d_1 + alpha |
|-1    a_22  -1            0         0      0   | | v_2   |   | h^2d_2         |
| 0    -1    a_33 . . .    0         0      0   | | v_3   |   | h^2d_3         |
```

```
 .                                                                          .
 .                                                                          .
 .                                                                          .
| 0     0     0   . . .  -1  a_(n-1)(n-1) -1    | | v_n-1 |    | h^2d_n-1        |
| 0     0     0   . . .   0          -1     a_nn | | v_n   |    | h^2d_n + beta   |
```

where $a_{ii} = \left(2 + h^2 c_i\right)$. This is a tridiagonal matrix, and can be easily solved using the Gaussian Elimination routine for tridiagonal matrices.

Example with a Neumann boundary condition

The example above is a little more complicated if the gradient at one or more of the boundaries is specified, instead of the values at the boundaries.

$$\begin{aligned} \frac{d^2 v(x)}{dx^2} &= c(x)v(x) + d(x);\ for\ 0 \le x \le 1 \\ & \qquad subject\ to\ v'(0) = \gamma; v'(1) = \delta; \end{aligned} \tag{8}$$

You still get the same general equation, $-v_{i-1} + \left(2 + h^2 c_i\right) v_i - v_{i+1} \ = \ h^2 d_i$, for $i = 0, \cdots, n+1$. To apply the gradient (Neumann) boundary conditions, you could use the forward- and backward-difference formulations at the endpoints. This would introduce a higher error into the calculations, however. An alternative is to use "fake" grid-cells at $0 - h$, and $1 + h$. These lie outside the model grid; but can be eliminated from the statement of the boundary conditions, and the general equations. The boundary conditions become

$$\begin{aligned} \frac{v_1 - v_{-1}}{2h} &= \gamma; \\ \frac{v_{n+2} - v_n}{2h} &= \delta; \end{aligned} \tag{9}$$

or, $v_{-1} = v_1 - 2h\gamma$, and $v_{n+2} = v_n + 2h\delta$. These can be used to eliminate v_{-1} and v_{n+2} from the equations centered about x_0 and x_{n+1}.

Periodic boundary conditions

You may want to use periodic boundary conditions, where $v(0) = v(1)$; or $v_0 = v_{n+1}$. This is done by setting $v_{-1} = v_n$ in the equation centered on x_0, and $v_{n+1} = v_0$ in the equation centered on x_n. This destroys the tri-diagonal structure of the equations, but the system can usually be solved.

Error in solution

Errors come into the finite difference method in two ways; rounding error in solving the set of equations $Av = d$, and discretization error in the approximations of v"(x) and v'(x). Ignoring the rounding error, the local error would equal $LDE \ = \ abs(v(x_i) - v_i)$, where $v(x_i)$ is the exact solution at $x = x_i$, and v_i is the approximation obtained using finite differences. The global discretization error, GDE, can be thought of as the error in the overall solution. Both the local and global errors are of order $O(h^2)$ if centered differences are used for a second order linear ordinary differential equation.

Shooting method

For a 2^{nd} order b.v.p. (boundary value problem), the **shooting method** uses one of the two boundary values specified. A guess is made at a second boundary condition (usually a combination of $v(x)$ and $v'(x)$ are used for x located at one of the boundaries, and the problem is solved using a method for ordinary differential equations. Often, a Runge-Kutta method would be used at this step. The value obtained for the second (as yet unused) boundary condition is compared to the value specified, and the error is used to guess another value of the unspecified boundary condition. The third guess can be a linear combination of the first two; weighted so as to give the exact value for the second boundary value that was specified. For linear boundary value problems, the third guess will be exact, so that only three iterations are required.