When the beam is of uniform thickness, the product EI is constant, and the exact solution is easily obtained. In many applications, however, the thickness is not uniform, so the moment of inertia I is a function of x, and approximation techniques are required. Problems of this type are considered in Exercises 7 of Section 11.3 and 6 of Section 11.4.

Methods for finding approximate solutions to differential equations, studied in Chapter 5, require that all conditions imposed on the differential equation occur at an initial point. For a second-order equation, we need to know both $w(0)$ and $w'(0)$, which is not the case in this problem. New techniques are required for handling problems when the conditions imposed are of a boundary-value rather than an initial-value type.

Physical problems that are position-dependent rather than time-dependent are often described in terms of differential equations with conditions imposed at more than one point. The two-point boundary-value problems in this chapter involve a second-order differential equation of the form

(11.1) $$y'' = f(x, y, y'), \quad a \le x \le b,$$

together with the boundary conditions

(11.2) $$y(a) = \alpha \quad \text{and} \quad y(b) = \beta.$$

11.1 The Linear Shooting Method

The following theorem gives general conditions that ensure that the solution to a second-order boundary value problem exists and is unique. The proof of this theorem can be found in [K,H].

Theorem 11.1 Suppose the function f in the boundary-value problem

$$y'' = f(x, y, y'), \quad a \le x \le b, \quad y(a) = \alpha, \quad y(b) = \beta,$$

is continuous on the set

$$D = \{(x, y, y') \mid a \le x \le b, -\infty < y < \infty, \ -\infty < y' < \infty\},$$

and that $\partial f/\partial y$ and $\partial f/\partial y'$ are also continuous on D. If

(i) $\dfrac{\partial f}{\partial y}(x, y, y') > 0$ for all $(x, y, y') \in D$, and

(ii) A constant M exists, with

$$\left| \frac{\partial f}{\partial y'}(x, y, y') \right| \le M, \quad \text{for all } (x, y, y') \in D,$$

then the boundary-value problem has a unique solution. ■

EXAMPLE 1 The boundary-value problem

$$y'' + e^{-xy} + \sin y' = 0, \quad 1 \le x \le 2, \quad y(1) = y(2) = 0,$$

has

$$f(x, y, y') = -e^{-xy} - \sin y'.$$

Since

$$\frac{\partial f}{\partial y}(x, y, y') = xe^{-xy} > 0 \quad \text{and} \quad \left|\frac{\partial f}{\partial y'}(x, y, y')\right| = |-\cos y'| \le 1,$$

this problem has a unique solution. ▪

When $f(x, y, y')$ has the form

$$f(x, y, y') = p(x)y' + q(x)y + r(x),$$

the differential equation

$$y'' = f(x, y, y')$$

is **linear**. Problems of this type frequently occur, and in this situation Theorem 11.1 can be simplified.

Corollary 11.2 If the linear boundary-value problem

$$y'' = p(x)y' + q(x)y + r(x), \quad a \le x \le b, \quad y(a) = \alpha, \quad y(b) = \beta$$

satisfies

(i) $p(x)$, $q(x)$, and $r(x)$ are continuous on $[a, b]$,
(ii) $q(x) > 0$ on $[a, b]$,

then the problem has a unique solution. ▪

To approximate the unique solution guaranteed by the satisfaction of the hypotheses of Corollary 11.2, let us first consider the initial-value problems

(11.3) $$y'' = p(x)y' + q(x)y + r(x), \quad a \le x \le b, \quad y(a) = \alpha, \quad y'(a) = 0,$$

and

(11.4) $$y'' = p(x)y' + q(x)y, \quad a \le x \le b, \quad y(a) = 0, \quad y'(a) = 1.$$

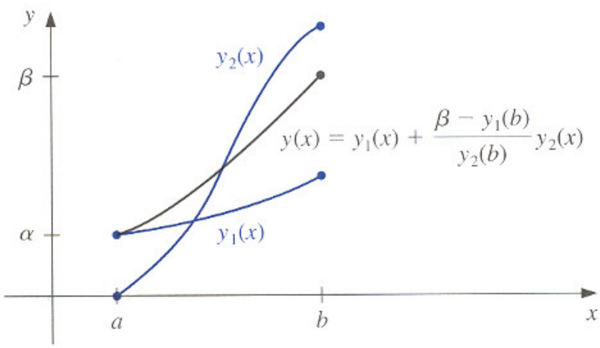
Theorem 5.16 in Section 5.9 ensures that under the hypotheses in Corollary 11.2, both problems have a unique solution. If $y_1(x)$ denotes the solution to (11.3) and $y_2(x)$ denotes the solution to (11.4), it is not difficult to verify that

$$y(x) = y_1(x) + \frac{\beta - y_1(b)}{y_2(b)} y_2(x) \tag{11.5}$$

is the unique solution to our boundary-value problem, provided, of course that $y_2(b) \neq 0$. (That $y_2(b) = 0$ is in conflict with the hypotheses of Corollary 11.2 is considered in Exercise 8.)

The Shooting method for linear equations is based on the replacement of the linear boundary-value problem by the two initial-value problems (11.3) and (11.4). Numerous methods are available from Chapter 5 for approximating the solutions $y_1(x)$ and $y_2(x)$, and once these approximations are available, the solution to the boundary-value problem is approximated using Eq. (11.5). Graphically, the method has the appearance shown in Figure 11.1.

gure 11.1



Algorithm 11.1 uses the fourth-order Runge-Kutta technique to find the approximations to $y_1(x)$ and $y_2(x)$, but any other technique for approximating the solutions to initial-value problems can be substituted into Step 4.

The algorithm has the additional feature of obtaining approximations for the derivative of the solution to the boundary-value problem as well as to the solution of the problem itself. The use of the algorithm is not restricted to those problems for which the hypotheses of Corollary 11.2 can be verified; it gives satisfactory results for many problems that do not satisfy these hypotheses.

ALGORITHM **11.1**

Linear Shooting

To approximate the solution of the boundary-value problem

$$-y'' + p(x)y' + q(x)y + r(x) = 0, \quad a \le x \le b, \quad y(a) = \alpha, \quad y(b) = \beta :$$

(Note: Equations (11.3) *and* (11.4) *are written as first-order systems and solved.)*

INPUT endpoints a, b; boundary conditions α, β; number of subintervals N.

OUTPUT approximations $w_{1,i}$ to $y(x_i)$; $w_{2,i}$ to $y'(x_i)$ for each $i = 0, 1, \ldots, N$.

Step 1 Set $h = (b - a)/N$;
$u_{1,0} = \alpha$;
$u_{2,0} = 0$;
$v_{1,0} = 0$;
$v_{2,0} = 1$.

Step 2 For $i = 0, \ldots, N - 1$ do Steps 3 and 4.
(The Runge-Kutta method for systems is used in Steps 3 *and* 4.)

Step 3 Set $x = a + ih$.

Step 4 Set $k_{1,1} = hu_{2,i}$;

$$k_{1,2} = h\left[p(x)u_{2,i} + q(x)u_{1,i} + r(x)\right];$$
$$k_{2,1} = h\left[u_{2,i} + \tfrac{1}{2}k_{1,2}\right];$$
$$k_{2,2} = h\left[p(x + h/2)\left(u_{2,i} + \tfrac{1}{2}k_{1,2}\right) + q(x + h/2)\left(u_{1,i} + \tfrac{1}{2}k_{1,1}\right) + r(x + h/2)\right];$$
$$k_{3,1} = h\left[u_{2,i} + \tfrac{1}{2}k_{2,2}\right];$$
$$k_{3,2} = h\left[p(x + h/2)\left(u_{2,i} + \tfrac{1}{2}k_{2,2}\right) + q(x + h/2)(u_{1,i} + \tfrac{1}{2}k_{2,1}) + r(x + h/2)\right];$$
$$k_{4,1} = h\left[u_{2,i} + k_{3,2}\right];$$
$$k_{4,2} = h\left[p(x + h)(u_{2,i} + k_{3,2}) + q(x + h)(u_{1,i} + k_{3,1}) + r(x + h)\right];$$
$$u_{1,i+1} = u_{1,i} + \tfrac{1}{6}\left[k_{1,1} + 2k_{2,1} + 2k_{3,1} + k_{4,1}\right];$$
$$u_{2,i+1} = u_{2,i} + \tfrac{1}{6}\left[k_{1,2} + 2k_{2,2} + 2k_{3,2} + k_{4,2}\right];$$
$$k'_{1,1} = hv_{2,i};$$
$$k'_{1,2} = h\left[p(x)v_{2,i} + q(x)v_{1,i}\right];$$
$$k'_{2,1} = h\left[v_{2,i} + \tfrac{1}{2}k'_{1,2}\right];$$
$$k'_{2,2} = h\left[p(x + h/2)\left(v_{2,i} + \tfrac{1}{2}k'_{1,2}\right) + q(x + h/2)\left(v_{1,i} + \tfrac{1}{2}k'_{1,1}\right)\right];$$
$$k'_{3,1} = h\left[v_{2,i} + \tfrac{1}{2}k'_{2,2}\right];$$
$$k'_{3,2} = h\left[p(x + h/2)\left(v_{2,i} + \tfrac{1}{2}k'_{2,2}\right) + q(x + h/2)\left(v_{1,i} + \tfrac{1}{2}k'_{2,1}\right)\right];$$
$$k'_{4,1} = h\left[v_{2,i} + k'_{3,2}\right];$$
$$k'_{4,2} = h\left[p(x + h)(v_{2,i} + k'_{3,2}) + q(x + h)(v_{1,i} + k'_{3,1})\right];$$
$$v_{1,i+1} = v_{1,i} + \tfrac{1}{6}\left[k'_{1,1} + 2k'_{2,1} + 2k'_{3,1} + k'_{4,1}\right];$$
$$v_{2,i+1} = v_{2,i} + \tfrac{1}{6}\left[k'_{1,2} + 2k'_{2,2} + 2k'_{3,2} + k'_{4,2}\right].$$

Step 5 Set $w_{1,0} = \alpha$;

$$w_{2,0} = \frac{\beta - u_{1,N}}{v_{1,N}};$$

OUTPUT $(a, w_{1,0}, w_{2,0})$.

Step 6 For $i = 1, \ldots, N$
set $W1 = u_{1,i} + w_{2,0}v_{1,i}$;
$W2 = u_{2,i} + w_{2,0}v_{2,i}$;
$x = a + ih$;
OUTPUT $(x, W1, W2)$. *(Output is $x_i, w_{1,i}, w_{2,i}$.)*

Step 7 STOP. *(Process is complete.)*

EXAMPLE 2 The boundary-value problem

$$y'' = -\frac{2}{x}y' + \frac{2}{x^2}y + \frac{\sin(\ln x)}{x^2}, \quad 1 \le x \le 2, \quad y(1) = 1, \quad y(2) = 2$$

has the exact solution

$$y = c_1 x + \frac{c_2}{x^2} - \frac{3}{10}\sin(\ln x) - \frac{1}{10}\cos(\ln x),$$

where

$$c_2 = \frac{1}{70}[8 - 12\sin(\ln 2) - 4\cos(\ln 2)] \approx -0.03920701320$$

and

$$c_1 = \frac{11}{10} - c_2 \approx 1.1392070132.$$

Applying Algorithm 11.1 to this problem requires approximating the solutions to the initial-value problems

$$y_1'' = -\frac{2}{x}y_1' + \frac{2}{x^2}y_1 + \frac{\sin(\ln x)}{x^2}, \quad 1 \le x \le 2, \quad y_1(1) = 1, \quad y_1'(1) = 0,$$

and

$$y_2'' = -\frac{2}{x}y_2' + \frac{2}{x^2}y_2, \quad 1 \le x \le 2, \quad y_2(1) = 0, \quad y_2'(1) = 1.$$

The results of the calculations, using Algorithm 11.1 with $N = 10$ and $h = 0.1$, are given in Table 11.1. The value listed as $u_{1,i}$ approximates $y_1(x_i)$, $v_{1,i}$ approximates $y_2(x_i)$, and w_i approximates

$$y(x_i) = y_1(x_i) + \frac{2 - y_1(2)}{y_2(2)} y_2(x_i). \qquad \blacksquare$$

The accuracy found in Table 11.1 is expected because the fourth-order Runge-Kutta method gives $O(h^4)$ accuracy to the solutions of the initial-value problems. Unfortunately,

Table 11.1

x_i	$u_{1,i}$	$v_{1,i}$	w_i	$y(x_i)$	$\lvert y(x_i) - w_i \rvert$
1.0	1.00000000	0.00000000	1.00000000	1.00000000	
1.1	1.00896058	0.09117986	1.09262917	1.09262930	1.43×10^{-7}
1.2	1.03245472	0.16851175	1.18708471	1.18708484	1.34×10^{-7}
1.3	1.06674375	0.23608704	1.28338227	1.28338236	9.78×10^{-8}
1.4	1.10928795	0.29659067	1.38144589	1.38144595	6.02×10^{-8}
1.5	1.15830000	0.35184379	1.48115939	1.48115942	3.06×10^{-8}
1.6	1.21248372	0.40311695	1.58239245	1.58239246	1.08×10^{-8}
1.7	1.27087454	0.45131840	1.68501396	1.68501396	5.43×10^{-10}
1.8	1.33273851	0.49711137	1.78889854	1.78889853	5.05×10^{-9}
1.9	1.39750618	0.54098928	1.89392951	1.89392951	4.41×10^{-9}
2.0	1.46472815	0.58332538	2.00000000	2.00000000	

because of roundoff errors, there can be problems hidden in this technique. If $y_1(x)$ rapidly increases as x goes from a to b, then $u_{1,N} \approx y_1(b)$ will be large. Should β be small in magnitude compared to $u_{1,N}$, the term $w_{2,0} = (\beta - u_{1,N})/v_{1,N}$ will be approximately $-u_{1,N}/v_{1,N}$. The computations in Step 6 then become

$$W1 = u_{1,i} + w_{2,0}v_{1,i} \approx u_{1,i} - \left(\frac{u_{1,N}}{v_{1,N}}\right) v_{1,i},$$

$$W2 = u_{2,i} + w_{2,0}v_{2,i} \approx u_{2,i} - \left(\frac{u_{1,N}}{v_{1,N}}\right) v_{2,i},$$

which allows a possibility of a loss of significant digits due to cancellation. However, since $u_{1,i}$ is an approximation to $y_1(x_i)$, the behavior of y_1 can easily be monitored, and if $u_{1,i}$ increases rapidly from a to b, the shooting technique can be employed backward, that is, solving instead the initial-value problems

$$y'' = p(x)y' + q(x)y + r(x), \quad a \le x \le b, \quad y(b) = \beta, \quad y'(b) = 0,$$

and

$$y'' = p(x)y' + q(x)y, \quad a \le x \le b, \quad y(b) = 0, \quad y'(b) = 1.$$

If this reverse shooting technique still gives cancellation of significant digits and if increased precision does not yield greater accuracy, other techniques must be used, such as those presented later in this chapter. In general, however, if $u_{1,i}$ and $v_{1,i}$ are $O(h^n)$ approximations to $y_1(x_i)$ and $y_2(x_i)$, respectively, for each $i = 0, 1, \ldots, N$, then $w_{1,i}$ will be an $O(h^n)$ approximation to $y(x_i)$. In particular,

$$|w_{1,i} - y(x_i)| \le Kh^n \left| 1 + \frac{v_{1,i}}{v_{1,N}} \right|,$$

for some constant K (see [IK, p. 426]).