DELFT UNIVERSITY OF TECHNOLOGY
Faculty of Electrical Engineering, Mathematics and Computer Science

ANSWERS OF THE TEST NUMERICAL METHODS FOR DIFFERENTIAL EQUATIONS (WI3097 TU)
Tuesday 31 January 2006, 14:00-17:00

1. a) In (1), substitute $u^\star$ into the second equation, replace u_n by the exact solution y_n, and expand

$$\overline{u_{n+1}} = y_n + \beta h f(t_n + \alpha h, y_n + \alpha h f(t_n, y_n))$$

into a Taylorseries:

$$\begin{aligned}\overline{u_{n+1}} &= y(t_n) + \beta h[f + \alpha h f_t + \alpha h f f_y + O(h^2)](t_n, y_n)\\ &= y(t_n) + \beta h f(t_n, y_n) + \alpha\beta h^2[f_t + f f_y](t_n, y_n) + O(h^3)\\ &= y(t_n) + \beta h y'(t_n) + \alpha\beta h^2 y''(t_n) + O(h^3),\end{aligned}$$

where $y'' = f_t + f f_y$ (an identity which follows by differentiation of the differential equation $y' = f(t, y)$) has been used to obtain the last line. Now subtracting the expansion $y(t_{n+1}) = y(t_n) + hy'(t_n) + \frac{h^2}{2}y''(t_n) + O(h^3)$ and dividing by h gives the expansion of the local truncation error τ_{n+1}:

$$\tau_{n+1} = \frac{y(t_{n+1}) - \overline{u_{n+1}}}{h} = (1-\beta)y(t_n) + (\frac{1}{2} - \alpha\beta)hy''(t_n) + O(h^2).$$

It follows that τ_{n+1} is O(1) unless β=1 and, furthermore, that α has to be chosen $\frac{1}{2}$ in order to achieve $\tau_{n+1} = \text{O}(h^2)$.

b) Substitute the test equation $y' = \lambda y$ into (1) with β=1:

$$\begin{aligned}u^\star &= u_n + \alpha h\lambda u_n = (1 + \alpha h\lambda)u_n\\ u_{n+1} &= u_n + h\lambda(1 + \alpha h\lambda)u_n = (1 + h\lambda + \alpha(h\lambda)^2)u_n.\end{aligned}$$

It follows that the amplification factor is $C(h\lambda) = 1 + h\lambda + \alpha(h\lambda)^2$.

c) The eigenvalues of the system follow by putting the characteristic polynomial

$$\det\begin{pmatrix}-\lambda & 1\\ -1 & -\lambda\end{pmatrix} = \lambda^2 + 1$$

equal to zero. This yields the two purely imaginary eigenvalues $\pm i$. Substitution of these eigenvalues into the amplification factor of b) gives $C(h\lambda) = 1 \pm ih - \alpha h^2$. For stability, the modulus of $C(h\lambda)$ should be smaller than 1:

$$(1 - \alpha h^2)^2 + h^2 < 1,$$

and thus

$$1 - 2\alpha h^2 + \alpha^2 h^4 + h^2 < 1.$$

This can be reduced to

$$-2\alpha + \alpha^2 h^2 + 1 < 0,$$

and rewritten as the stabiltity condition

$$h^2 < \frac{2\alpha - 1}{\alpha^2}.$$

It follows that no stable stepsizes exist for $\alpha < \frac{1}{2}$.

d) Putting $x_1 = y$ en $x_2 = y'$, we have $x_1' = x_2$, while substitution of the new names into $y'' + 6y' + 5y = \sin t$ gives $x_2' + 6x_2 + 36x_1 = \sin t$. In vector form:

$$\mathbf{x}' = \begin{pmatrix} 0 & 1 \\ -36 & -6 \end{pmatrix} \mathbf{x} + \begin{pmatrix} 0 \\ \sin t \end{pmatrix}. \qquad (1)$$

Now put

$$\det \begin{pmatrix} -\lambda & 1 \\ -36 & -6-\lambda \end{pmatrix} = \lambda^2 + 6\lambda + 36 = (\lambda + 3)^2 + 27$$

equal to zero and it follows that $-3 \pm 3i\sqrt{3}$ are the eigenvalues of the system (1).

e) Putting β=1 and $\alpha = 0$ in (1) we recover the forward Euler method with the stability region as given in the hint of part e) of the assignment. Now mark the eigenvalues in the complex $h\lambda$-domain. A rough estimate then shows that a stepsize of $\frac{1}{10}$ is sufficiently small to bring $h\lambda$ into the Euler stability region. The precise upperbound for h ($\frac{1}{6}$) is somewhat larger than 0.1.

2. (a) The exact answer is 0.75. The composite Trapezoidal rule is given by

$$\frac{1}{2} \cdot \{\frac{1}{2} \cdot (1 - 0^3) + 1 - (\frac{1}{2})^3 + \frac{1}{2} \cdot (1 - 1^3)\} = \frac{11}{16} = 0.6875.$$

The difference with the exact answer is 0.0625.

(b) The rounding error is less than

$$h \cdot \{\frac{1}{2}\epsilon + \epsilon \ldots + \epsilon + \frac{1}{2}\epsilon\} \leq n \cdot h \cdot \epsilon = (b - a) \cdot \epsilon.$$

(c) The Taylor polynomial is given by

$$P_1(x) = f(a) + (x - a)f'(a)$$

whereas the truncation error is:

$$f(x) - P_1(x) = \frac{(x-a)^2}{2} f''(\xi), \text{ with } \xi \in [a, b].$$

(d) Integrating this formula gives:

$$\int_a^b P_1(x)dx = \int_a^b f(a) + (x-a)f'(a)dx = (b-a)f(a) + \frac{(b-a)^2}{2} f'(a).$$

Suppose that $M_2 = \max_{\xi \in [a,b]} |f''(\xi)|$. This implies that $|f(x) - P_1(x)| \leq \frac{(x-a)^2}{2} M_2$. Integrating this formula gives:

$$|\int_a^b f(x)dx - \left((b-a)f(a) + \frac{(b-a)^2}{2} f'(a)\right) \leq \int_a^b |f(x) - P_1(x)|dx \leq$$

$$\int_a^b \frac{(x-a)^2}{2} M_2 dx = \frac{(b-a)^3}{6} M_2$$

(e) The composite rule is:

$$h \cdot \{f(a) + \frac{h}{2} f'(a) + f(a+h) + \frac{h}{2} f'(a+h) \ldots + f(b-h) + \frac{h}{2} f'(b-h)\}.$$

The result with the composite rule is:

$$\frac{1}{2} \cdot \{(1-0^3) - 3 \cdot \frac{1}{2} \cdot \frac{1}{2} \cdot 0^2 + (1 - (\frac{1}{2})^3) - 3 \cdot \frac{1}{2} \cdot \frac{1}{2} \cdot (\frac{1}{2})^2\} = \frac{27}{32} = 0.8438.$$

The difference with the exact answer is 0.0938.

(f) For the comparison we note that

- the new method has a worse behavior with respect to rounding errors, because rounding errors of f' also play a role.
- the new method costs n function evaluations (of f') more than the Trapezoidal rule
- The truncation error of the new method is given by

$$\frac{n \cdot h^3}{6} \max_{\xi \in [a,b]} |f''(\xi)| = \frac{(b-a)h^2}{6} \max_{\xi \in [a,b]} |f''(\xi)|$$

  which is 2 times as large as the truncation error of the Trapezoidal rule.

Conclusion: the new method is worse than the Trapezoidal rule.