Mathematical Tripos Part IB: Lent 2010

Numerical Analysis – Lecture 10[1]

Definition We say that the ODE $\boldsymbol{y}' = \boldsymbol{f}(t, \boldsymbol{y})$ is *stiff* if (for some methods) we need to depress h to maintain *stability* well beyond requirements of accuracy. An important example of stiff systems occurs when an equation is linear, $\mathrm{Re}\,\lambda_l < 0$, $l = 1, 2, \ldots, N$, and the quotient $\max|\lambda_k|/\min|\lambda_k|$ is large: a ratio of 10^{20} is not unusual in real-life problems!

Stiff equations, mostly nonlinear, occur throughout applications, whenever we have two (or more) different timescales in the ODE. A typical example are equations of *chemical kinetics,* where each timescale is determined by the speed of reaction between two compounds: such speeds can differ by many orders of magnitude.

Definition Suppose that a numerical method, applied to $y' = \lambda y$, $y(0) = 1$, with constant h, produces the solution sequence $\{y_n\}_{n\in\mathbb{Z}^+}$. We call the set

$$\mathcal{D} = \{h\lambda \in \mathbb{C} : \lim_{n\to\infty} y_n = 0\}$$

the *linear stability domain* of the method. Noting that the set of $\lambda \in \mathbb{C}$ for which $y(t) \overset{t\to\infty}{\longrightarrow} 0$ is the left half-plane $\mathbb{C}^- = \{z \in \mathbb{C} : \mathrm{Re}\, z < 0\}$, we say that the method is *A-stable* if $\mathbb{C}^- \subseteq \mathcal{D}$.

Example We have already seen that for Euler's method $y_n \to 0$ iff $|1 + h\lambda| < 1$, therefore $\mathcal{D} = \{z \in \mathbb{C} : |1 + z| < 1\}$. Moreover, solving $y' = \lambda y$ with the *trapezoidal rule,* we obtain $y_{n+1} = [(1 + \frac{1}{2}h\lambda)/(1 - \frac{1}{2}h\lambda)]y_n$ thus, by induction, $y_n = [(1 + \frac{1}{2}h\lambda)/(1 - \frac{1}{2}h\lambda)]^n y_0$. Therefore

$$z \in \mathcal{D} \qquad \Leftrightarrow \qquad \left|\frac{1 + \frac{1}{2}z}{1 - \frac{1}{2}z}\right| < 1 \qquad \Leftrightarrow \qquad \mathrm{Re}\, z < 0$$

and we deduce that $\mathcal{D} = \mathbb{C}^-$. Hence, the method is A-stable.
It can be proved by similar means that for *backward Euler* it is true that $\mathcal{D} = \{z \in \mathbb{C} : |1 - z| > 1\}$, hence that the method is also A-stable.

Note that A-stability does not mean that *any* step size will do! We need to choose h small enough to ensure the right accuracy, but we don't want to depress it much further to prevent instability.

Discussion A-stability analysis of multistep methods is considerably more complicated. However, according to the *second Dahlquist barrier,* no multistep method of order $p \geq 3$ may be A-stable. Note that the $p = 2$ barrier for A-stability is attained by the trapezoidal rule.

The Dahlquist barrier implies that, in our quest for higher-order methods with good stability properties, we need to pursue one of the following strategies:

- either relax the definition of A-stability
- or consider other methods in place of multistep.

The two courses of action will be considered next.

Stiffness and BDF methods Inasmuch as no multistep method of order $p \geq 3$ may be A-stable, stability properties of BDF, say, are satisfactory for most stiff equations. The point is that in many stiff linear systems in applications the eigenvalues are not just in $\mathbb{C}^-$ but also well away from $\mathrm{i}\mathbb{R}$. *[Analysis of nonlinear stiff equations is difficult and well outside the scope of this course.]* All BDF methods of order $p \leq 6$ (i.e., all convergent BDF methods) share the feature that the linear stability domain $\mathcal{D}$ includes a wedge about $(-\infty, 0)$: such methods are said to be A_0*-stable.*

Stiffness and Runge–Kutta Unlike multistep methods, implicit high-order RK may be A-stable. For example, recall the 3rd-order method

$$\boldsymbol{k}_1 = \boldsymbol{f}\left(t_n, \boldsymbol{y}_n + \tfrac{1}{4}h(\boldsymbol{k}_1 - \boldsymbol{k}_2)\right),$$

[1]Corrections and suggestions to these notes should be emailed to `A.Iserles@damtp.cam.ac.uk`. All handouts are available on the WWW at the URL `http://www.damtp.cam.ac.uk/user/na/PartIB/Handouts.html`.

$$
\begin{aligned}
\boldsymbol{k}_2 &= \boldsymbol{f}\left(t_n + \tfrac{2}{3}h, \boldsymbol{y}_n + \tfrac{1}{12}h(3\boldsymbol{k}_1 + 5\boldsymbol{k}_2)\right), \\
\boldsymbol{y}_{n+1} &= \boldsymbol{y}_n + \tfrac{1}{4}h(\boldsymbol{k}_1 + 3\boldsymbol{k}_2).
\end{aligned}
$$

from the last lecture. Applying it to $y' = \lambda y$, we have

$$
\begin{aligned}
hk_1 &= h\lambda\left(y_n + \tfrac{1}{4}hk_1 - \tfrac{1}{4}hk_2\right), \\
hk_2 &= h\lambda\left(y_n + \tfrac{1}{4}hk_1 + \tfrac{5}{12}hk_2\right).
\end{aligned}
$$

This is a linear system, whose solution is

$$
\begin{bmatrix} hk_1 \\ hk_2 \end{bmatrix} = \begin{bmatrix} 1 - \frac{1}{4}h\lambda & \frac{1}{4}h\lambda \\ -\frac{1}{4}h\lambda & 1 - \frac{5}{12}h\lambda \end{bmatrix}^{-1} \begin{bmatrix} h\lambda y_n \\ h\lambda y_n \end{bmatrix} = \frac{h\lambda y_n}{1 - \frac{2}{3}h\lambda + \frac{1}{6}(h\lambda)^2} \begin{bmatrix} 1 - \frac{2}{3}h\lambda \\ 1 \end{bmatrix},
$$

therefore

$$
y_{n+1} = y_n + \tfrac{1}{4}hk_1 + \tfrac{3}{4}hk_2 = \frac{1 + \frac{1}{3}h\lambda}{1 - \frac{2}{3}h\lambda + \frac{1}{6}h^2\lambda^2} y_n.
$$

Let

$$
r(z) = \frac{1 + \frac{1}{3}z}{1 - \frac{2}{3}z + \frac{1}{6}z^2}.
$$

Then $y_{n+1} = r(h\lambda)y_n$, therefore, by induction, $y_n = [r(h\lambda)]^n y_0$ and we deduce that

$$
\mathcal{D} = \{z \in \mathbb{C} : |r(z)| < 1\}
$$

We wish to prove that $|r(z)| < 1$ for every $z \in \mathbb{C}^-$, since this is equivalent to A-stability. This will be done by a technique that can be applied to other RK methods. According to the *maximum modulus principle* from Complex Methods, if g is analytic in the closed complex domain $\mathcal{V}$ then $|g|$ attains its maximum on $\partial\mathcal{V}$. We let $g = r$. This is a rational function, hence its only singularities are the poles $2 \pm \mathrm{i}\sqrt{2}$ and g is analytic in $\mathcal{V} = \mathrm{cl}\,\mathbb{C}^- = \{z \in \mathbb{C} : \mathrm{Re}\, z \le 0\}$. Therefore it attains its maximum on $\partial\mathcal{V} = \mathrm{i}\mathbb{R}$ and

$$
\text{A-stability} \quad \Leftrightarrow \quad |r(z)| < 1, \quad z \in \mathbb{C}^- \quad \Leftrightarrow \quad |r(\mathrm{i}t)| \le 1, \quad t \in \mathbb{R}.
$$

In turn,

$$
|r(\mathrm{i}t)|^2 \le 1 \quad \Leftrightarrow \quad |1 - \tfrac{2}{3}\mathrm{i}t - \tfrac{1}{6}t^2|^2 - |1 + \tfrac{1}{3}\mathrm{i}t|^2 \ge 0.
$$

But $|1 - \frac{2}{3}\mathrm{i}t - \frac{1}{6}t^2|^2 - |1 + \frac{1}{3}\mathrm{i}t|^2 = \frac{1}{36}t^4 \ge 0$ and it follows that the method is A-stable.

Example It is possible to prove that the 2-stage *Gauss–Legendre method*

$$
\begin{aligned}
\boldsymbol{k}_1 &= \boldsymbol{f}(t_n + (\tfrac{1}{2} - \tfrac{\sqrt{3}}{6})h, \boldsymbol{y}_n + \tfrac{1}{4}h\boldsymbol{k}_1 + (\tfrac{1}{4} - \tfrac{\sqrt{3}}{6})h\boldsymbol{k}_2), \\
\boldsymbol{k}_2 &= \boldsymbol{f}(t_n + (\tfrac{1}{2} + \tfrac{\sqrt{3}}{6})h, \boldsymbol{y}_n + (\tfrac{1}{4} + \tfrac{\sqrt{3}}{6})h\boldsymbol{k}_1 + \tfrac{1}{4}h\boldsymbol{k}_2), \\
\boldsymbol{y}_{n+1} &= \boldsymbol{y}_n + \tfrac{1}{2}h(\boldsymbol{k}_1 + \boldsymbol{k}_2)
\end{aligned}
$$

is of order 4. [You can do this for $y' = f(y)$ by expansion, but it becomes messy for $\boldsymbol{y}' = \boldsymbol{f}(t, \boldsymbol{y})$.] It can be easily verified that for $y' = \lambda y$ we have $y_n = [r(h\lambda)]^n y_0$, where $r(z) = (1 + \frac{1}{2}z + \frac{1}{12}z^2)/(1 - \frac{1}{2}z + \frac{1}{12}z^2)$. Since the poles of r reside at $3 \pm \mathrm{i}\sqrt{3}$ and $|r(\mathrm{i}t)| \equiv 1$, we can again use the maximum modulus principle to argue that $\mathcal{D} = \mathbb{C}^-$ and the Gauss–Legendre method is A-stable.

4.5 Implementation of ODE methods

The step size h is not some preordained quantity: it is a parameter of the method (in reality, many parameters, since we may vary it from step to step). The basic input of a well-written computer package for ODEs is not the step size but the *error tolerance:* the level of precision, as required by the user. The choice of $h > 0$ is an important tool at our disposal to keep a local estimate of the error beneath the required tolerance in the solution interval. In other words, we need not just a *time-stepping algorithm,* but also mechanisms for *error control* and for amending the step size.