Lecture 5
Initial Value Problem for Ordinary Differential Equations Local and Global Errors: Stability

1. Error Analysis of Linear Multistep Methods

Consider a linear multistep method in general form

$$a_k y_n + a_{k-1} y_{n-1} + ... + a_0 y_{n-k} = h[b_k f_n + b_{k-1} f_{n-1} + ... + b_0 f_{n-k}] \quad (1)$$

for solving an initial value problem

$$y' = f(x, y), \qquad y(x_0) = x_0. \quad (2)$$

This is called a k-step method. The coefficients a_i, b_i are given. As before, y_i denotes an approximation to the solution at $x_i = x_0 + ih$, and $f_i = f(x_i, y_i)$. This formula is used to compute y_n, assuming that $y_{n-k}, y_{n-k+1}, ..., y_{n-1}$ are known. We assume that $a_k \neq 0$. If $b_k = 0$, the method is said to be explicit, and y_n can be computed directly. In the opposite case the method is said to be implicit. The accuracy of a numerical solution is largely determined by the order of the algorithm used. The order indicates how many terms in a Taylor series solution are being simulated by the method. The following linear functional determines the order of the algorithm:

$$L(y) = \sum_{i=0}^{n} [a_i y(x_0 + ih) - h b_i y'(x_0 + ih)]. \quad (3)$$

Using Taylor series for $y(ih)$ and $y'(ih)$ with $i = 0, 1, ..., k$

$$y(x_0 + ih) = \sum_{j=0}^{+\infty} \frac{(ih)^j}{j!} y^{(j)}(x_0), \qquad y'(x_0 + ih) = \sum_{j=0}^{+\infty} \frac{(ih)^j}{j!} y^{(j+1)}(x_0),$$

we obtain

$$L(y) = d_0 y(x_0) + d_1 h y'(x_0) + d_2 h^2 y''(x_0) + \dots , \tag{4}$$

where

$$d_0 = \sum_{i=0}^{k} a_i, \qquad d_1 = \sum_{i=0}^{k} (i a_i - b_i), \qquad d_2 = \sum_{i=0}^{k} (i^2 a_i/2 - i b_i), \dots$$

$$d_j = \sum_{i=0}^{k} \left(\frac{i^j}{j!} a_i - \frac{i^{j-1}}{(j-1)!} b_i \right).$$

If $d_0 = d_1 = \dots = d_m = 0$, then

$$L(y) = d_{m+1} h^{m+1} y^{(m+1)}(x_0) + O(h^{m+2})$$

represents local truncation error, and the method has the order m. For example, in the case

$$y_n - y_{n-2} = \frac{1}{3} h (f_n + 4 f_{n-1} + f_{n-2})$$

we have a_1, a_2, a_3 is $-1, 0, 1$, and b_1, b_2, b_3 is $1/3, 4/3, 1/3$, and

$$d_0 = a_0 + a_1 + a_2 = 0, \qquad d_1 = -b_0 + (a_1 - b_1) + (2a_2 - b_2) = 0,$$

$$d_2 = d_3 = d_4 = 0, \qquad d_5 = -1/90.$$

Hence, the order of the method is 4, and local truncation error is of order $O(h^5)$.

2. Stability and Consistency

Consider again a multistep method

$$a_k y_n + a_{k-1} y_{n-1} + \dots + a_0 y_{n-k} = h[b_k f_n + b_{k-1} f_{n-1} + \dots + b_0 f_{n-k}] \tag{5}$$

for solving an initial value problem

$$y' = f(x, y), \qquad y(x_0) = x_0. \tag{6}$$

Associated with the multistep method are two polynomials

$$p(z) = a_k z^k + a_{k-1} z^{k-1} + ... + a_0, \qquad q(z) = b_k z^k + b_{k-1} z^{k-1} + ... + b_0.$$

The analysis will show that certain desirable properties of the multistep method depend on the location of the roots of the polynomials $p(z)$ and $q(z)$.

Convergence of the method means that the approximate solution $y(h, x)$ obtained by using step size h satisfies

$$\lim_{h \to 0} y(h, x) = y(x)$$

for fixed $x = x_0 + nh$ $(n \to +\infty)$, provided that the starting value obeys the same initial condition

$$\lim_{h \to 0} y(h, x_0 + nh) = y_0$$

for n fixed. The method is **stable** if all roots of $p(z)$ lie in the circle $|z| \leq 1$ and each root of modulus 1 is simple. The method is **consistent** if $p(1) = 0$ and $p'(1) = q(1)$.

The theorem states: for the multistep method to be convergent, it is necessary and sufficient that it be stable and consistent.

We shall demonstrate only that stability and consistency are necessary. Let us assume that there is a root $|\lambda| > 1$. Consider the initial value problem

$$y' = 0, \qquad y(0) = 0.$$

Then the solution is governed by

$$a_k y_n + a_{k-1} y_{n-1} + ... + a_0 y_{n-k} = 0.$$

This is a linear difference equation. We seek solution in the form $y_n = hz^n$. Then, we obtain for z characteristic equation $p(z) = 0$. If $z = \lambda$ such that $|\lambda| > 1$, the corresponding solution behaves like this

$$|y(h, x_0 + nh)| = |h\lambda^n| < h|\lambda|^n \to +\infty$$

for fixed $x = x_0 + nh$ as $h \to 0$ and $n \to +\infty$. This divergent partial solution will result in divergence of a corresponding numerical solution. If $|\lambda| = 1$ and $p'(\lambda) = 0$, then solution is $nh\lambda^n$, again we have

$$|y(h, x_0 + nh)| = |hn\lambda^n| < hn|\lambda|^n \to +\infty$$

for fixed $x = x_0 + nh$ as $h \to 0$ and $n \to +\infty$.

Why consistency is necessary? Consider the initial value problem

$$y' = 0, \qquad y(0) = 1.$$

The solution is $y = 1$, and $y_n = 1$ all arbitrary n. On the other side the governing equation is the same. Thus, it must be $p(1) = 0$. Consider the initial value problem

$$y' = 1, \qquad y(0) = 0.$$

The exact solution is $y = x$ and $y_n = hn$. The governing equation becomes

$$a_k y_n + a_{k-1} y_{n-1} + ... + a_0 y_{n-k} = h[b_k + b_{k-1} + ... + b_0].$$

Hence, it is necessary that for $n = k$

$$a_k k + a_{k-1}(k-1) + ... + a_1 = b_k + b_{k-1} + ... + b_0,$$

that is $p'(1) = q(1)$.

Example - Milne Method

The Milne Method is defined by

$$y_n - y_{n-2} = \frac{h}{3}(f_n + 4f_{n-1} + f_{n-2}).$$

This is implicit method and it is characterised by

$$p(z) = z^2 - 1, \qquad q(z) = (z^2 + 4z + 1)/3.$$

The simple roots of $p(z)$ are ± 1. Furthermore, $p'(z) = 2z$, $p'(1) = q(1) = 2$. Thus, the conditions of stability and consistency are fulfilled. By theorem, the Milne Method is convergent.

3. Global Truncation Error

Consider the initial value problem

$$x' = f(t, x), \qquad x(0) = x_0, \qquad 0 \leq t \leq T > 0.$$

How to establish a bound on the global truncation error in solving a initial problem for a differential equation? We assume that no round error is involved. The difference $x(t_n) - x_n$ is the global truncation error. It is not simply the sum of all local truncation errors that entered at the previous points. The key point is to understand how two solutions differ at any point if they are started with different initial conditions as each step in the numerical solution must use as its initial value the approximated ordinate computed at the preceding step. Consider the initial value problem

$$x' = f(t, x), \qquad x(0) = s, \qquad 0 \leq t \leq T > 0.$$

We assume that $f_x = \frac{\partial f}{\partial x}(t,x)$ is continuous and satisfies the condition $f_x \leq \lambda$ for $0 \leq t \leq T > 0$. The solution is $x = x(t,s)$. We would like to know how the solution depends on s. Define $u(t) = \frac{\partial x}{\partial s}(t,s)$. We can obtain a differential equation - the variational equation - for u by differentiating with respect to s in the initial value problem

$$u'(t) = f_x(t,x)u, \qquad u(0) = 1, \qquad 0 \leq t \leq T > 0.$$

Note that if $f_x \leq \lambda$ for $0 \leq t \leq T > 0$, then the solution of the variational equation satisfies inequality

$$|u(t)| \leq e^{\lambda t}, \qquad 0 \leq t \leq T > 0.$$

Proof: we have

$$u'/u = f_x = \lambda - \alpha(t), \qquad \alpha(t) \geq 0.$$

Integrating this inequality, we obtain

$$\log|u| = \lambda t - \int_0^t \alpha(\tau)d\tau,$$

$$|u(t)| = \exp\left\{\lambda t - \int_0^t \alpha(\tau)d\tau\right\} \leq e^{\lambda t},$$

because of

$$\int_0^t \alpha(\tau)d\tau \geq 0.$$

Using this inequality, it is easily to show that if the initial value problem is solved with two initial values s and $s+\delta$, the solutions differ at t by at most $|\delta|e^{\lambda t}$, as

$$|x(t,s) - x(t,s+\delta)| = \left|\frac{\partial}{\partial s}x(t,s+\theta\delta)\right||\delta| = |u(t)||\delta| \leq e^{\lambda t}|\delta|, \qquad 0 < \theta < 1.$$

Global Error Theorem: if all local truncation errors are $t_1, t_2, ..., t_n$ do not exceed δ in magnitude, then the global truncation error does nor exceed $\delta(1 - e^{n\lambda h})(1 - e^{\lambda h})^{-1}$.

Proof: In computing x_1 there was an error $|\delta_1|$. In computing x_2 the global error is

$$|\delta_1|e^{\lambda h} + |\delta_2|,$$

where the first term in the right-hand side is the error in the initial condition, and the second term is the new truncation error. In computing x_3 the global error is

$$(|\delta_1|e^{\lambda h} + |\delta_2|)e^{\lambda h} + |\delta_3|,$$

and so on. Finally, if $|\delta_i| \leq \delta$, $i = 1, 2, ..., n$, we obtain for the global truncation error

$$|\delta_1|e^{n\lambda h} + |\delta_2|e^{(n-1)\lambda h} + ... + |\delta_n| \leq \delta\frac{1 - e^{n\lambda h}}{1 - e^{\lambda h}}.$$

If all local truncation errors $\delta_i = O(h^{m+1})$, then the global truncation error is $O(h^m)$.

4. Weak stability

The results of the last section imply that we can assume that our linear multistep method is convergent (consistent and zero-stable). We consider the problem

$$x' = -x,\ x(0) = 1, \tag{7}$$

whose exact solution is $x(t) = e^{-t}$. We are going to compare the exact solution to the approximate solution. By way of example, let us solve this

problem using the $k = 2$ Adams-Bashforth method. The numerical solution satisfies the recurrence formula:

$$x_{n+2} - x_{n+1} = -h\left(\frac{3}{2}x_{n+1} - \frac{1}{2}x_n\right). \tag{8}$$

We could also ask what happens to *errors* e_n in the solution – do they increase or decrease? The errors satisfy the same equation as the solution itself (note only in this case!):

$$e_{n+2} - e_{n+1} = -h\left(\frac{3}{2}e_{n+1} - \frac{h}{2}e_n\right). \tag{9}$$

We first ask about absolute stability. The solution is absolutely stable if errors e_n decrease. We suppose, as usual, that $e_n = \lambda e_{n-1}$. We then plug into (9) and cancel through by e_n, obtaining:

$$\lambda^2 + \left(\frac{3h}{2} - 1\right)\lambda - \frac{h}{2} = 0. \tag{10}$$

This equation has two solutions, λ_1 and λ_2, with

$$\lambda_1 = \left(-\frac{3h}{4} + \frac{1}{2}\right) + \frac{1}{2}\sqrt{1 - h + \frac{9h^2}{4}}; \tag{11}$$

$$\lambda_2 = \left(-\frac{3h}{4} + \frac{1}{2}\right) - \frac{1}{2}\sqrt{1 - h + \frac{9h^2}{4}}. \tag{12}$$

The general solution is that

$$e_n = A\lambda_1^n + B\lambda_2^n. \tag{13}$$

So long as $|\lambda_{1,2}| \leq 1$, $|e_n|$ will decrease with n and hence with t, and the solution will be absolutely stable. For $h \in [0, 1]$, the stability condition $|\lambda_{1,2}| \leq 1$ is satisfied, and the solutions are stable.

Is this enough? Well, actually *NO*! Let us instead solve for x_n. We have the same equations for x_n as for e_n. So in this easy case, the solution will similarly be:

$$x_n = P\lambda_1^n + Q\lambda_2^n. \tag{14}$$

Let us compare this solution to the actual solution, by translating solutions in terms of λ into solutions as a function of t. We have $t = nh$, and in the limit of small h we can do a whole load of Taylor expanding to obtain:

$$\begin{aligned}
\lambda_1 &= \left(-\frac{3h}{4} + \frac{1}{2}\right) + \frac{1}{2}\sqrt{1 - h + \frac{9h^2}{4}} \\
&\approx \left(-\frac{3h}{4} + \frac{1}{2}\right) + \frac{1}{2}\left(1 - \frac{h}{2} + \frac{9h^2}{8} - \frac{h^2}{8}\right) \\
&\approx 1 - h + \frac{1}{2}h^2 + O(h^3) \\
&\approx \exp(-h) + O(h^3).
\end{aligned} \tag{15}$$

Anyway we can ignore the $O(h^3)$ term for the moment. Thus

$$\lambda_1^n \approx \exp(-nh) = \exp(-t)\left(1 + O(h^3)\right). \tag{16}$$

Clearly this is the solution to the algorithm which corresponds to the real solution of the differential equation. But wait! There is also the λ_2 solution. We see that $|\lambda_2| < |\lambda_1|$ for small h, and so when iterated it rapidly disappears relative to the real solution. There are two solutions: a *physical* and a *spurious* solution. For low h, the physical solution dominates the spurious solution. (In fact for small h, we can also expand λ_2 in a Taylor series, to obtain $\lambda_2 \approx \frac{h}{2}$.) At $h \neq 0$, of course, the physical solution of the difference equation will not be exactly right, but it will tend to the exact solution. But when h increases, the absolute value of the spurious root increases, and the absolute

value of the physical root decreases. So long as $|\lambda_1| > |\lambda_2|$, eventually all evidence of the spurious solution disappears for long times (t $\sim n \to \infty$), for

$$\frac{|\lambda_2|}{|\lambda_1|} < 1 \Rightarrow \lim_{n\to\infty} \frac{|\lambda_2^n|}{|\lambda_1^n|} = 0. \tag{17}$$

However, for sufficiently large h the ratio reaches unity, and calculated explicitly, at $h = h_c = \frac{2}{3}$. Once $h > h_c$, the spurious root will dominate the physical solution. The solution we will find numerically will look *nothing like* the real solution. If we were to believe the algorithm with $h > h_c$, we get a completely wrong idea of the solution, *even though the algorithm is absolutely stable*. The relative error will increase. The solution, although absolutely stable, is *relatively unstable*. The interval of relative stability for h is $[0, \frac{2}{3}]$, but the interval of absolute stability was $[0, 1]$. We have only scratched the surface of stability theory in these notes. However, this example is enough to see that all sorts of complications can show up when considering stability.

5. High Order ODE and System of ODE

The initial value problem for n-th order ODE looks like this

$$y^{(n)} = f(x, y, y', y'', ..., y^{(n-1)}), \quad y = y_0, \;\; y' = y'_0, ..., y^{(n-1)} = y_0^{(n-1)} \text{at } x = x_0.$$

The initial value problem for a system of ODE looks like this

$$\begin{cases} y'_1 = f_1(x, y_1, y_2, ..., y_n), \\ y'_2 = f_2(x, y_1, y_2, ..., y_n), \\ ....., \\ y'_n = f_n(x, y_1, y_2, ..., y_n), \end{cases}$$

and at $x = x_0$ we have $y_1 = y_1^{(0)}, ..., y_n = y_n^{(0)}$. A n-th order ODE can be converted to a system of n first-order equations:

$$\begin{cases} y'_1 = y_2, \\ y'_2 = y_3, \\ ....., \\ y'_n = f_n(x, y_1, y_2, ..., y_n), \end{cases}$$

and at x_0 the initial conditions are like that $y_1 = y_0, \;\; y_2 = y'_0, ..., \; y_n = y_0^{(n-1)}$. Consider a special type of system of ODE

$$\begin{cases} \dot{x}_1(t) = f_1(x_1, x_2, ..., x_n), \\ \dot{x}_2(t) = f_2(x_1, x_2, ..., x_n), \\ ....., \\ \dot{x}_n(t) = f_n(x_1, x_2, ..., x_n), \end{cases}$$

and at $t = t_0$ we have the initial values $x_1 = x_1^{(0)}, ..., x_n = x_n^{(0)}$. A system in the form is said to be autonomous. This is well-known in mechanics where

$x_1, ..., x_n$ mean coordinates of mechanical system and t means a time. The system can be written in the vector form

$$\dot{X} = F(X),$$

where the unknown vector X is $(x_1(t), ..., x_n(t))^T$. The Runge-Kutta procedure for a system of first order equations is most easily written down in the case when our system is autonomous

$$X(t+h) = X(t) + \frac{1}{6}(F_1 + 2F_2 + 2F_3 + F_4),$$

where

$$F_1 = hF(X), \quad F_2 = hF(X+\frac{1}{2}F_1), \quad F_3 = hF(X+\frac{1}{2}F_2), \quad F_4 = hF(X+F_3).$$

Multistep methods can also be extended to apply to systems of equations. As an example, we give the vector form of the Adams-Bashforth-Moulton predictor-corrector method for the autonomous system

$$X^*(t+h) = X(t) + \frac{h}{720}[1901F(X(t))-$$

$$2774F(X(t-h)) + 2616F(X(t-2h)) - 1274F(X(t-3h)) + 251(X(t-4h))],$$

$$X(t+h) = X(t) + \frac{h}{720}[251F^*(X(t+1))+$$

$$646F(X(t)) - 264F(X(t-h)) + 106F(X(t-2h)) - 19(X(t-3h))].$$

As in the case of a single equation, a single-step procedure such as a fifth-order Runge-Kutta method could be used to provide starting values:

$$X(t_0+h), \quad X(t_0+2h), \quad X(t_0+3h), \quad X(t_0+4h).$$