Solving Ordinary Differential equations

Taylor methods can be used to build explicit methods with higher order of convergence than Euler's method. The main difficult of these methods is the computation of the derivatives of $f(t, y)$.

The idea behind these methods is simple: considering the Taylor series of $y(t_{n+1})$ up to a certain order. For example, by truncating up to order h^2 we will have the Taylor method of order 2.

Let us build this method. Consider:

$$y(t_{n+1}) = y(t_n) + hy'(t_n) + \frac{h^2}{2}y''(t_n) + \frac{h^3}{6}y'''(\zeta_n)$$

We will take up to order h^2. For computing $y'(t_n)$ and $y''(t_n)$ we will use the differential equation:

$$\begin{aligned}
&y'(t) = f(t, y(t)) \\
&y''(t) = f_t(t, y(t)) + f_y(t, y(t))y'(t) = f_t(t, y(t)) + f_y(t, y(t))f(t, y(t))
\end{aligned}$$

Therefore we have the method

$$y_{n+1} = y_n + hf(t_n, y_n) + \frac{h^2}{2}(f_t(t_n, y_n) + f_y(t, y_n)f(t_n, y_n))$$

Solving Ordinary Differential equations

By defining as the local truncation error the term which is neglected for building the method, we see that in our method this error is $T_n = \frac{h^3}{6} y'''(\zeta_n)$, which satisfies $||T_n|| \le Ch^3$. This means that the method is order 2.

Error Analysis

1. Truncation Error (Truncating Taylor Series) [Large step size $\Longrightarrow$ large errors]
2. Rounding Error (Machine precision) [very small step size $\Longrightarrow$ roundoff errors]

Two kinds of Truncation of Error:

1. **Local** - error within one step due to application of method
2. **Propagation** - error due to previous local errors.

Global Truncation Error = Local + Propagation

Generally if local truncation error is $\mathcal{O}(h^{(n+1)})$ then, the Global truncation error is $\mathcal{O}(h^{(n)})$.

Solving Ordinary Differential equations

General Single-step methods:
Before introducing the Runge-Kutta methods, we will briefly describe the general set-up for Single-step methods. In general, these methods can be written as:

$$y_{n+1} = y_n + h\Phi(x_n, y_n, y_{n+1}, h) \tag{1}$$

where Φ is related to f and it is called the Step Function. When Φ depends on y_{n+1} the method is implicit and in other case the method is explicit.

Single-Step Methods for I.V. Problems

The examples of single-step methods considered so far are:

1. Forward Euler, $\Phi(x_n, y_n, y_{n+1}, h) = f(x_n, y_n)$ (explicit).
2. Trapezoidal, $\Phi(x_n, y_n, y_{n+1}, h) = (f(x_n, y_n) + f(x_{n+1}, y_{n+1}))/2$ (implicit).
3. Taylor of order 2, $\Phi(x_n, y_n, y_{n+1}, h) = f(t_n, y_n) + \frac{h}{2}(f_t(t_n, y_n) + f_y(t, y_n)f(t_n, y_n)$ (explicit).

Single-Step Methods for I.V. Problems

For Single-step methods

Definition

We define the local truncation error, T_n, as:

$$T_n(h) = y(t_{n+1}) - y(t_n) - h\Phi(t_n, y(t_n), y(t_{n+1}), h) \tag{2}$$

Single-Step Methods for I.V. Problems

Runge-Kutta methods are very popular methods which allow us to obtain high-order methods avoiding the computation of the derivatives of the Taylor methods.
Let's see how to build an explicit Runge-Kutta method of order 2:

$$\begin{aligned} y_{n+1} &= y_n + h\Phi(t_n, y_n, h) \\ \Phi(t_n, y_n, h) &= ak_1 + bf(t_n + \alpha h, y_n + \beta h k_1)\,,\ k_1 = f(t_n, y_n) \end{aligned} \tag{3}$$

The constants of the Step function will be determine by imposing that the method behaves like the Taylor method up to order 2. The Taylor method of order 2 was:

$$y_{n+1} = y_n + hf(t_n, y_n) + \frac{h^2}{2}(f_t(t_n, y_n) + f_y(t, y_n)f(t_n, y_n))$$

Then, in order to compare with the new method, we just need to expand $f(t_n + \alpha h, y_n + \beta h k_1)$ up to order 1:

$$\begin{aligned} f(t_n + \alpha h, y_n + \beta h k_1) &= f(t_n, y_n) + f_t(t_n, y_n)\alpha h + f_y(t_n, y_n)\beta h k_1 + \mathcal{O}(h^2) \\ &= f(t_n, y_n) + f_t(t_n, y_n)\alpha h + f_y(t_n, y_n)f(t_n, y_n)\beta h + \mathcal{O}(h^2 \end{aligned}$$

Single-Step Methods for I.V. Problems

Going to Eq. (3), we have that up to order 2 the new method is:

$$y_{n+1} = y_n + h(a+b)f(t_n, y_n) + bh^2(\alpha f_t(t_n, y_n) + \beta f(t_n, y_n)f_y(t_n y_n))$$

Then, if the method must behave as the Taylor method up to order 2, we have the following equations for the parameters of the method:

$$a + b = 1\,,\ \ b\alpha = b\beta = 1/2$$

The methods verifying these equations will be convergents with order of convergence 2.
Consider, for instance, the methods with $\alpha = \beta$. Then, $b = 1/2\alpha$ and $a = 1 - 1/2\alpha$. The methods with $\alpha = 1/2$ and $\alpha = 1$ are called the Modified Euler Method and the Heun Method, respectively.

Single-Step Methods for I.V. Problems

The Modified Euler Method ($\alpha = 1/2$) can be written as

$$\boxed{y_{n+1} = y_n + hf(t_n + \frac{h}{2}, y_n + \frac{h}{2} f_n)\,. \qquad (4)}$$

and using the notation

$$k_1 = f(t_n, y_n)\,, \quad k_2 = f(t_n + \frac{h}{2}, y_n + \frac{h}{2} k_1)\,,$$

we have that

$$y_{n+1} = y_n + hk_2\,.$$

Single-Step Methods for I.V. Problems

The Heun Method ($\alpha = 1$), can be written as

$$\boxed{y_{n+1} = y_n + \frac{h}{2}\left(f(t_n, y(t_n)) + f(t_{n+1}, y_n + hf(t_n, y_n))\right) . \qquad (5)}$$

If

$$k_1 = f(t_n, y_n)\,, \quad k_2 = f(t_n + h, y_n + hk_1)\,,$$

then, we have:

$$y_{n+1} = y_n + \frac{h}{2}(k_1 + k_2)\,.$$

Single-Step Methods for I.V. Problems

The previous notation can be generalized

$$k_1 = f(t_n, y_n), \;\; k_2 = f(t_n + c_2 h, y_n + a_{21} h k_1)\,,$$

and

$$y_{n+1} = y_n + h(b_1 k_1 + b_2 k_2)\,,$$

where c_2, a_{21}, b_1 and b_2 are constants.
All these methods are members of a big family of methods which are called the Runge-Kutta Methods.

Single-Step Methods for I.V. Problems

The general Runge-Kutta *s*-Stages Method is

$$y_{n+1} = y_n + h \sum_{i=1}^{s} b_i k_i \,,$$

where

$$k_i = f\left(t_n + c_i h, y_n + h \sum_{j=1}^{s} a_{ij} k_j \right) \,, \quad i = 1, 2, ..., s.$$

A convenient way of displaying the coefficients of the Runge-Kutta Methods is the use of the Butcher tableaux:

$$\begin{array}{c|c} \boldsymbol{c} & A \\ \hline & \boldsymbol{b}^T \end{array} = \begin{array}{c|cccccc} c_1 & a_{11} & a_{12} & a_{13} & \dots & a_{1s} \\ c_2 & a_{21} & a_{22} & a_{23} & \dots & a_{2s} \\ c_3 & a_{31} & a_{32} & a_{33} & \dots & a_{3s} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ c_s & a_{s1} & a_{s2} & a_{s3} & \dots & a_{ss} \\ \hline & b_1 & b_2 & b_3 & \dots & b_s \end{array}$$

Single-Step Methods for I.V. Problems

Example

Find the Butcher tableaux of the following Runge-Kutta Method:

$$y_{n+1} = y_n + h\left(\frac{1}{4}k_1 + \frac{3}{4}k_2\right)$$

$$k_1 = f\left(t_n, y_n + h\left(\frac{1}{4}k_1 - \frac{1}{4}k_2\right)\right),$$

$$k_2 = f\left(t_n + \frac{2}{3}h, y_n + h\left(\frac{1}{4}k_1 + \frac{5}{12}k_2\right)\right).$$

Sol.:

$$\begin{array}{c|cc} 0 & \frac{1}{4} & -\frac{1}{4} \\ \frac{2}{3} & \frac{1}{4} & \frac{5}{12} \\ \hline & \frac{1}{4} & \frac{3}{4} \end{array}$$

Single-Step Methods for I.V. Problems

If $p^*(s)$ is the maximum order of convergence which can be obtained by using an explicit Runge-Kutta s-stages method, then we have

$$\begin{aligned}
p^*(s) &= s, \quad s = 1, 2, 3, 4 \\
p^*(5) &= 4 \\
p^*(6) &= 5 \\
p^*(7) &= 6 \\
p^*(8) &= 6 \\
p^*(9) &= 7 \\
p^*(s) &\leq s - 2\,, \quad s = 10, 11, \ldots
\end{aligned}$$

This behaviour explains the popularity of the 4-stages Runge-Kutta Methods of order 4.

Single-Step Methods for I.V. Problems

Adaptive Step-size Control
Goal: with little additional effort estimate (bound) the magnitude of local truncation error at each step so that step size can be reduced/increased if local error increases/decreases.
Basic idea: 2. Use a matched pair of Runge-Kutta formulas of order p and $p + 1$ that use common values of ki, and yield estimate or bound local truncation error.

Single-Step Methods for I.V. Problems

R-K Method of Order p:

$$y_{n+1} = y_n + h\sum_{i=1}^{s} b_i k_i$$

R-K method of Order $p+1$:

$$y_{n+1} = y_n + h\sum_{i=1}^{s} \tilde{b}_i k_i$$

These methods are called embedded Runge-Kutta Methods and can be expressed by using the following modified Butcher tableaux:

$$\begin{array}{c|c} \mathbf{c} & A \\ \hline & \mathbf{b}^T \\ & \tilde{\mathbf{b}}^T \\ \hline & \mathbf{b}^T - \tilde{\mathbf{b}}^T \end{array}$$

Single-Step Methods for I.V. Problems

The local truncation error for the order-p method can be estimated as follows:

$$\tilde{T}_{n+1} = h \sum_{i=1}^{s} (b_i - \tilde{b}_i) k_i \,.$$

Then, the step-size can be adapted by imposing the following condition:

$$\tilde{T}_{n+1} < Tol,$$

being *Tol* the tolerance by step unit.
The “new” step size is then selected as follows:

$$h_{new} = \left(\frac{q Tol}{|\tilde{T}_{n+1}|} \right)^{\frac{1}{p+1}} h \,. \qquad (6)$$

where q is a factor with a value of $q \approx 0.8$.

Single-Step Methods for I.V. Problems

Example: The Runge-Kutta-Fehlberg (4,5) scheme has the following Butcher tableaux

$$
\begin{array}{c|cccccc}
0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\frac{1}{4} & \frac{1}{4} & 0 & 0 & 0 & 0 & 0 \\
\frac{3}{8} & \frac{3}{32} & \frac{9}{32} & 0 & 0 & 0 & 0 \\
\frac{12}{13} & \frac{1932}{2197} & -\frac{7200}{2197} & \frac{7296}{2197} & 0 & 0 & 0 \\
1 & \frac{439}{216} & -8 & \frac{3680}{513} & -\frac{845}{4104} & 0 & 0 \\
\frac{1}{2} & -\frac{8}{27} & 2 & -\frac{3544}{2565} & \frac{1859}{4104} & -\frac{11}{40} & 0 \\
\hline
 & \frac{25}{216} & 0 & \frac{1408}{2565} & \frac{2197}{4104} & -\frac{1}{5} & 0 \\
 & \frac{16}{135} & 0 & \frac{6656}{12825} & \frac{28561}{56430} & -\frac{9}{50} & \frac{2}{55} \\
\hline
 & \frac{1}{360} & 0 & -\frac{128}{4275} & -\frac{2197}{75240} & \frac{1}{50} & \frac{2}{55}
\end{array}
$$

Single-Step Methods for I.V. Problems

Higher-Order ODEs and Systems of Equations
An $n-$th order ODE can be converged into a system of n coupled 1st-order ODEs. Systems of first order ODEs are solved just as one solves a single ODE.
Example:
Consider the 4th-order ODE

$$y'''' + a(x)y''' + b(x)y'' + c(x)y' + d(x)y = f(x)$$

By letting

$$y''' = v_3;\ \ y'' = v_2;\ \text{and}\ y' = v_1$$

this 4th-order ODE can be written as a system of four coupled 1st-order ODEs.

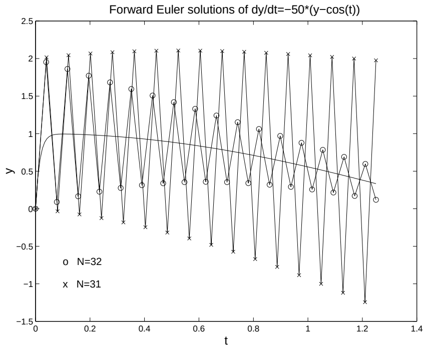
Single-Step Methods for I.V. Problems

Stiff Differential Equations
A stiff system of ODE's is one involving rapidly changing components together with slowly changing ones. In many cases, the rapidly varying components die away quickly, after which the solution is dominated by the slow ones.
Consider the following ODE:

$$y' = -50(y - \cos(t))\,, \;\; 0 \le t \le 1.25\,, \;\; y(0) = 0\,,$$

The figure shows the approximations obtained by using the Forward Euler method for $h = 1.25/31$ and $h = 1.25/32$.

Single-Step Methods for I.V. Problems



Single-Step Methods for I.V. Problems

In order to ensure the stability of the numerical solution, we have to choose an step size $h < 2/50$. This means that, for stiff problems, we would need a very small step to capture the behavior of the rapid transient and to preserve a stable and accurate solution, and this would then make it very laborious to compute the slowly evolving solution.

Euler's method is known as an explicit method because the derivative is taken at the known point i. An alternative is to use an implicit approach, in which the derivative is evaluated at a future step $i + 1$. The simplest method of this type is the backward Euler method that yields

$$y_{n+1} = y_n + hf(y_{n+1}, t_{n+1}) \, .$$

In general, for this kind of problems, suitable methods are the BDF *backward differentiation formulae* methods. These methods have the general form:

$$\frac{1}{h} \sum_{j=1}^{k} \frac{1}{j} \nabla^j y_{n+1} = f_{n+1} \, .$$

Linear Multistep Methods for I.V. Problems

The linear multistep methods can be written in the general form

$$\boxed{\sum_{j=0}^{k} \alpha_j y_{n+j} = h \sum_{j=0}^{k} \beta_j f_{n+j}\,, \qquad (7)}$$

where k is called the step number and without loss of generality we let $\alpha_k = 1$. Explicit methods are characterised by $\beta_k = 0$ and implicit methods have $\beta_k \neq 0$.

Linear Multistep Methods for I.V. Problems

The Adams Family

This familiy of methods is derived from the identity

$$y(t_{n+1}) - y(t_n) = \int_{t_n}^{t_{n+1}} f(t, y(t))dt\,. \tag{8}$$

With a view to using previously computed values of y_n, we replace $f(t, y)$ by the polynomial of degree $k - 1$ passing through the k points

$$(t_{n-k+1}, f_{n-k+1}),\ \ldots, (t_{n-1}, f_{n-1}), (t_n, f_n)\,.$$

Using a constant step size, this polynomial can be written (in the Newton backward difference form) as follows

$$P_{k-1}(r) = f_n + r\nabla f_n + \frac{r(r+1)}{2!}\nabla^2 f_n + \ldots + \frac{r(r+1)\ldots(r+k-2)}{(k-1)!}\nabla^{k-1} f_n\,.$$

donde $t = t_n + rh$, $r \in [-(k-1), 0]$.

Linear Multistep Methods for I.V. Problems

Then, the Adams-Bashforth method of k steps has the form

$$y_{n+1} - y_n = \int_{t_n}^{t_{n+1}} P_{k-1}(t)dt\,.$$

and by integrating the polynomial

$$y_{n+1} - y_n = \int_0^1 P_{k-1}(r)hdr = h\sum_{i=0}^{k-1} \gamma_i \nabla^i f_n = \sum_{i=0}^{k-1} \gamma_i \Delta^i f_{n-i} \qquad (9)$$

where $\gamma_i = (-1)^i \int_0^1 \left(\begin{array}{c} -r \\ i \end{array} \right) dr$

$$\gamma_0 = \int_0^1 dr = 1, \;\; \gamma_1 = \int_0^1 rdr = \frac{1}{2}, \;\; \gamma_2 = \int_0^1 \frac{r(r+1)}{2} dr = \frac{5}{2}\,,$$

$$\gamma_3 = \int_0^1 \frac{r(r+1)(r+2)}{6} dr = \frac{3}{8}\,, \;\; \dots$$

Linear Multistep Methods for I.V. Problems

The first Adams-Bashforth methods are then:

$$\begin{aligned}
y_{n+1} &= y_n + hf_n && \text{(AB1), Euler Method}\\
y_{n+1} &= y_n + \frac{h}{2}(3f_n - f_{n-1}) && \text{(AB2)}\\
y_{n+1} &= y_n + \frac{h}{12}(23f_n - 16f_{n-1} + 5f_{n-2}) && \text{(AB3)}\\
y_{n+1} &= y_n + \frac{h}{24}(55f_n - 59f_{n-1} + 37f_{n-2} - 9f_{n-3}) && \text{(AB4)}
\end{aligned} \tag{10}$$

and so on.

Linear Multistep Methods for I.V. Problems

The Adams-Bashforth methods are **explicit methods**. **Implicit Adams methods** are derived by integrating the order k polynomial passing through the points

$$(t_{n+1}, f_{n+1}),\ (t_n, f_n),\ (t_{n-1}, f_{n-1})\ldots, (t_{n-k+1}, f_{n-k+1})\,.$$

Note that we are considering the additional data point (t_{n+1}, f_{n+1}). The corresponding polynomial is

$$P_k(r) = f_{n+1} + r\nabla f_{n+1} + \frac{r(r+1)}{2!}\nabla^2 f_{n+1} + \ldots + \frac{r(r+1)\ldots(r+k-1)}{k!}\nabla^k f_{n+1}\,.$$

where $t = t_{n+1} + rh$, $r \in [-(k-1), 0]$.

Linear Multistep Methods for I.V. Problems

Then, for the implicit methods we will have

$$y_{n+1} - y_n = \int_{-1}^{0} P_k(r) h dr = h \sum_{i=0}^{k} \gamma_i' \nabla^i f_{n+1} = h \sum_{i=0}^{k} \gamma_i' \Delta^i f_{n+1-i} \, ,$$

where

$$\gamma_0' = \int_{-1}^{0} dr = 1, \;\; \gamma_1' = \int_{-1}^{0} r dr = -\frac{1}{2}, \;\; \gamma_2' = \int_{-1}^{0} \frac{r(r+1)}{2} dr = -\frac{1}{12} \, ,$$

$$\gamma_3' = \int_{-1}^{0} \frac{r(r+1)(r+2)}{6} dr = -\frac{1}{24} \, , \;\; \dots$$

Linear Multistep Methods for I.V. Problems

The implicit Adams methods are called Adams-Moulton methods (AM). The first AM methods are:

$$
\begin{aligned}
y_{n+1} &= y_n + hf_{n+1} = y_n + hf(t_{n+1} y_{n+1}) && \text{(AM0), implicit Euler method}\\
y_{n+1} &= y_n + \frac{h}{2}(f_{n+1} + f_n) && \text{(AM1), trapezoidal method}\\
y_{n+1} &= y_n + \frac{h}{12}(5f_{n+1} + 8f_n - f_{n-1}) && \text{(AM2)}\\
y_{n+1} &= y_n + \frac{h}{24}(9f_{n+1} - 19f_n - 5f_{n-1} + f_{n-2}) && \text{(AM3)}
\end{aligned}
\tag{11}
$$

Note that, as typical for any implicit method, we have to solve a nonlinear equation.