

Course: Numerical Solution of Ordinary Differential Equations

Module 2: Multi-step methods

<i>Lecture</i>	<i>Content</i>	<i>Hours</i>
1	Single step and Multi-step methods, Predictor corrector methods, Milne's method	2
2	Adams-Moulton method.	1
3	Adams Bashforth method	1

Module 2

Lecture 1

Multi Step Methods

Predictor corrector Methods

keywords: multi-step predictor, corrector, Milne-simpson method, integration formulae

Consider IVP

$$y' = f(t, y); \quad t_0 \leq t \leq b \quad \text{with} \quad y(t_0) = y_0 \quad (2.1)$$

One step methods for solving IVP (2.1) are those methods in which the solution y_{j+1} at the $j+1^{\text{th}}$ grid point involves only one previous grid point where the solution is already known. Accordingly, a general one step method may be written as

$$y_{j+1} = y_j + h\phi(t_j, y_j, h)$$

The increment function Φ depends on solution y_j at previous grid point t_j and step size h . If y_{j+1} can be determined simply by evaluating right hand side then the method is explicit method. The methods developed in the module 1 are one step methods. These methods might use additional functional evaluations at number of points between t_j and t_{j+1} . These functional evaluations are not used in further computations at advanced grid points. In these methods step size can be changed according to the requirement.

It may be reasonable to develop methods that use more information about the solution (functional values and derivatives) at previously known values while computing solution at the next grid point. Such methods using information at more than one previous grid points are known as multi-step methods and are expected to give better results than one step methods.

To determine solution y_{j+1} , a multi-step method or k -step method uses values of $y(t)$ and $f(t, y(t))$ at k previous grid points $t_{j-k}, k=0, 1, 2, \dots, k-1$. y_j is called the initial point while y_{j-k} are starting points. The starting points are computed using some suitable one step method. Thus multi-step methods are not self starting methods.

Integrating (2.1) over an interval (t_{j-k}, t_{j+1}) yields

$$y_{j+1} = y_j + \int_{t_{j-k}}^{t_{j+1}} f(t, y(t)) dt \cong y_j + \int_{t_{j-k}}^{t_{j+1}} \sum_{j=0}^r a_j t^j dt \quad (2.2)$$

The integrand on the right side is approximated by interpolating polynomial of degree r using equi-spaced points. The integration over the interval is shown in the Fig 2.1.

The method may be explicit or implicit. An implicit method involves computation of y_{j+1} in terms of y_{j+1} . First an explicit formula known as predictor formula is used to predict y_{j+1} . Then another formula, known as corrector formula, is used to improve the predicted

value of y_{j+1} . The predictor-corrector methods form a large class of general methods for numerical integration of ordinary differential equations. A popular predictor-corrector scheme is known as the Milne-Simpson method.

Milne-Simpson method

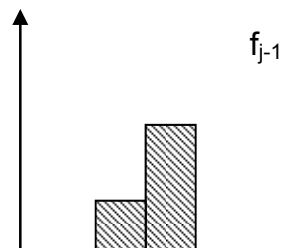
Its predictor is based on integration of $f(t, y(t))$ over the interval $[t_{j-3}, t_{j+1}]$ with $k=3$ and $r=3$. The interpolating polynomial is considered to match the function at three points t_{j-2} , t_{j-1} , and t_j and the function is extrapolated at both the ends in the interval $[t_{j-3}, t_{j-2}]$ and $[t_j, t_{j+1}]$ as shown in the Fig 2.2(a). Since the end points are not used, an open integration formula is used for the integral in (2.2):

$$p_{j+1} = y_{j+1} = y_j + \frac{4h}{3} (2f(t_j, y_j) - f(t_{j-1}, y_{j-1}) + 2f(t_{j-2}, y_{j-2})) + \frac{14}{45} h^5 f^{(4)}(\xi); \xi \text{ in } (t_{j-3}, t_{j+1}) \quad (2.3)$$

The explicit predictor formula is of $O(h^4)$ and requires starting values. These starting values should also be of same order of accuracy. Accordingly, if the initial point is y_0 then the starting values y_1 , y_2 and y_3 are computed by fourth order Runge kutta method. Then predictor formula (2.3) predicts the approximate solution y_4 as p_4 at next grid point. The predictor formula (2.3) is found to be unstable (proof not included) and the solution so obtained may grow exponentially.

The predicted value is then improved using a corrector formula. The corrector formula is developed similarly. For this, a second polynomial for $f(t, y(t))$ is constructed, which is based on the points (t_{j-1}, f_{j-1}) , (t_j, f_j) and the predicted point (t_{j+1}, f_{j+1}) . The closed integration of the interpolating polynomial over the interval $[t_j, t_{j+1}]$ is carried out [See Fig 2.2 (b)]. The result is the familiar Simpson's rule:

$$y_{j+1} = y_j + \frac{h}{3} (f(t_{j+1}, y_{j+1}) + 4f(t_j, y_j) + f(t_{j-1}, y_{j-1})) - \frac{1}{90} h^5 f^{(4)}(\xi); \xi \text{ in } (t_{j-1}, t_{j+1}) \quad (2.4)$$



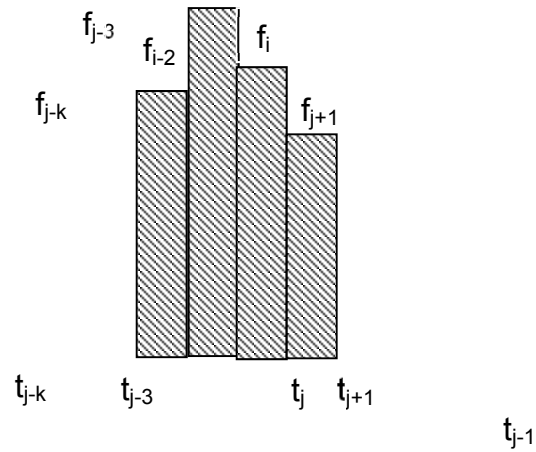


Fig 2.1 Scheme for multi-step integration

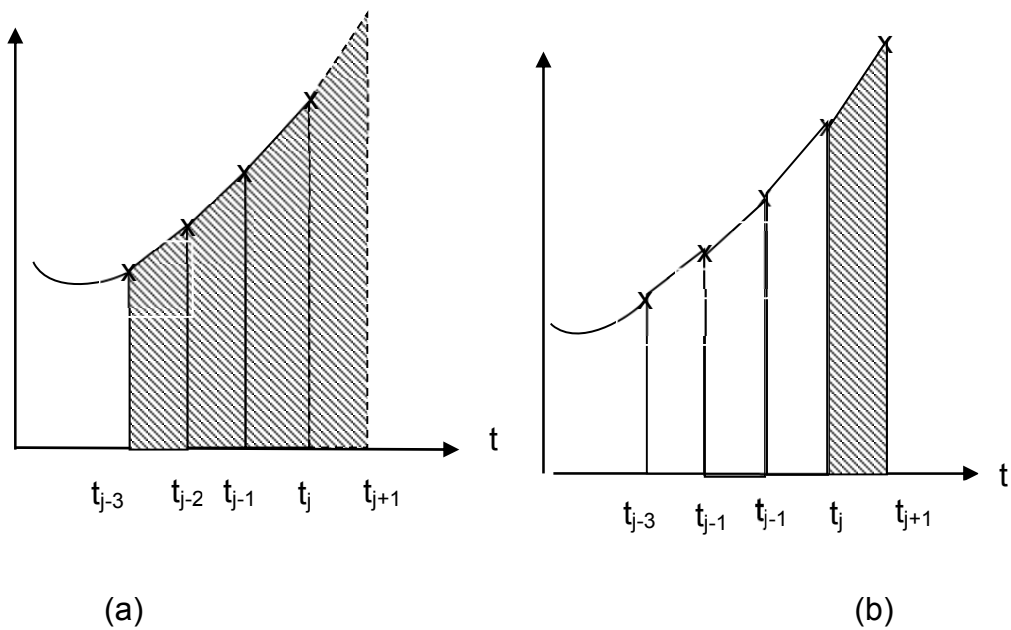


Fig 2.2 (a) Open Scheme for Predictor (b) Closed integration for Corrector

In the corrector formula f_{j+1} is computed from the predicted value p_{j+1} as obtained from (2.3).

Denoting $f_j = f(t_j, y_j)$, the equations (2.3) and (2.4) gives the following predictor corrector formulae, respectively, for solving IVP (2.1) at equi-spaced discrete points $t_4, t_5, \dots$

$$p_{j+1} = y_{j+1} = y_{j-3} + \frac{4h}{3} [2f_j - f_{j-1} + 2f_{j-2}] \quad (2.5)$$

$$y_{j+1} = y_{j-1} + \frac{h}{3} [f_{j+1} + 4f_j + f_{j-1}]$$

The solution at initial point t_0 is given in the initial condition and t_1, t_2 and t_3 are the starting points where solution is to be computed using some other suitable method such as Runge Kutta method. This is illustrated in the example 2.1

Example 2.1: Solve IVP $y' = y + 3t - t^2$ with $y(0) = 1$; using Milne's predictor corrector method take $h = 0.1$

Solution: The following table 2.1 computes Starting values using fourth order Runge Kutta method.

k	t	y	k1= f(t,y)	t+ h/2	y+ h/2*k	k2	y+ h/2*k2	k3	t+h	y+ h*k3	k4	y+h(k1+2k 2+2k3+k4) /6
0	0	1	1	0.05	1.05	1.1975	1.059875	1.2074	0.1	1.1207	1.41074	1.1203415
1	0.1	1.1203	1.41034	0.15	1.190859	0.2227	1.131477	1.559	0.2	1.2762	1.83624	1.2338409
2	0.2	1.2338	1.79384	0.25	1.323533	0.3218	1.249931	1.9374	0.3	1.4276	2.23758	1.3763387
3	0.2	1.3763										

Table 2.1: Starting values using RK4 in Example 2.1

Using initial value and starting values at $t = 0, 0.1, 0.2$ and 0.3 , the predictor formula predicts the solution at $t = 0.4$ as 1.7199359. It is used in corrector formula to give the corrected value. The solution is continued at advanced grid points [see table 2.2].

Milne Predictor corrector1					f(t,p)
k	t	y	f(t,y)	corrector	
0	0	1	1		
1	0.1	1.1203415	1.410341		
2	0.2	1.2338409	1.793841		
3	0.3	1.3763387	2.186339		
4	0.4	1.7199359	2.759936	1.67714525	2.717145
5	0.5	2.0317593	3.281759	1.920894708	3.170895

Table 2.2: Example 2.1 using predictor corrector method with $h=0.1$

The exact solution is possible in this example; however it may not be possible for other equations. Table 2.3 compares the solution with the exact solution of given equation. Clearly the accuracy is better in predictor corrector method than the Runge-Kutta method.

	rk4	Milne pc	exact
0.1	1.120341		1.120342
0.2	1.233841		1.282806
0.3	1.376339		1.489718
0.4	1.5452	1.6771453	1.743649
0.5	1.7369	1.9208947	2.047443

Table 2.3: Comparison of solution of example 2.1

	Milne	Predictor		corrector1	f(t,p)
k	t	y	f(t,y)		
0	0	1	1		
1	0.05	1.055042	1.202542	starting point1	
2	0.1	1.120342	1.410342	starting point2	
3	0.15	1.196169	1.623669	starting point3	
4	0.2	1.282805	1.842805	1.282805	1.842805
5	0.25	1.380551	2.068051	1.380551	2.068051

Table 2.4: Example 2.1 using predictor corrector method with $h=0.05$

The exercise 2.1 is repeated with $h=0.5$ in table 2.4. The table 2.5 clearly indicates that the better accuracy is achieved with $h=0.05$ [see table 2.5]

0.05	1.055042		1.055042
0.1	1.120342		1.120342
0.15	1.196169		1.196168
0.2		1.282805	1.282806
0.25		1.380551	1.380551

Table 2.5: improved accuracy with $h=0.05$

Predictor Corrector methods are preferred over Runge-Kutta as it requires only two functional evaluations per integration step while the corresponding fourth order Runge-Kutta requires four evaluations. The starting points are the weakness of predictor-corrector methods. In Runge kutta methods the step size can be changed easily.