6.4 Adams-Bashforth-Moulton methods

The most famous linear multistep methods are constructed by the means of interpolation. For instance by the following strategy:

The solution of the ODE satisfy the integral equation

\[ y(t_{n+1}) - y(t_n) = \int_{t_n}^{t_{n+1}} f(t, y(t)) dt. \]  \hfill (48)

Assume that we have found \( f_i = f(t_i, y_i) \) for \( i = n-k+1, \ldots, n \), with \( t_i = t_0 + ih \). Construct the polynomial of degree \( k-1 \), satisfying

\[ p_{k-1}(t_i) = f(t_i, y_i), \quad i = n-k+1, \ldots, n. \]

The interpolation points are equidistributed (constant stepsize), so Newton’s backward difference formula can be used in this case (see Exercise 2), that is

\[ p_{k-1}(t) = p_{k-1}(t_n + sh) = f_n + \sum_{j=1}^{k-1} (-1)^j \binom{-s}{j} \nabla^j f_n \]

where

\[ (-1)^j \binom{-s}{j} = \frac{s(s+1)\cdots(s+j-1)}{j!} \]

and

\[ \nabla^0 f_n = f_n, \quad \nabla^j f_n = \nabla^{j-1} f_n - \nabla^{j-1} f_{n-1}. \]

Using \( y_{n+1} \approx y(t_{n+1}) \), \( y_n \approx y(t_n) \) and \( p_{k-1}(t) \approx f(t, y(t)) \) in (48) gives

\[ y_{n+1} - y_n \int_{t_n}^{t_{n+1}} p_{k-1}(t) dt = h \int_{t_n}^{t_{n+1}} p_{k-1}(t_n + sh) ds \]

\[ = hf_n + h \sum_{j=1}^{k-1} \left( (-1)^j \int_0^1 \binom{-s}{j} ds \right) \nabla^j f_n. \]  \hfill (49)

This gives the Adams-Bashforth methods

\[ y_{n+1} - y_n = h \sum_{j=0}^{k-1} \gamma_j \nabla^j f_n, \quad \gamma_0 = 1, \quad \gamma_j = (-1)^j \int_0^1 \binom{-s}{j} ds. \]

Example 6.6. We get

\[ \gamma_0 = 1, \quad \gamma_1 = \int_0^1 sds = \frac{1}{2}, \quad \gamma_2 = \int_0^1 \frac{s(s+1)}{2} ds = \frac{5}{12} \]

and the first few methods becomes:

\[ y_{n+1} - y_n = hf_n \]

\[ y_{n+1} - y_n = h \left( \frac{3}{2} f_n - \frac{1}{2} f_{n-1} \right) \]

\[ y_{n+1} - y_n = h \left( \frac{23}{12} f_n - \frac{4}{3} f_{n-1} + \frac{5}{12} f_{n-2} \right) \]
A $k$-step Adams-Bashforth method is explicit, has order $k$ (which is the optimal order for explicit methods) and it is zero-stable. In addition, the error constant $C_{p+1} = \gamma_k$. Implicit Adams methods are constructed similarly, but in this case we include the (unknown) point $(t_{n+1}, f_{n+1})$ into the set of interpolation points. So the polynomial
\[ p^*_k(t) = p^*_k(t_n + sh) = f_{n+1} + \sum_{j=1}^{k} (-1)^j \binom{s+1}{j} \nabla^j f_{n+1} \]
interpolates the points $(t_i, f_i)$, $i = n-k+1, \ldots, n+1$. Using this, we get the Adams-Moulton methods
\[ y_{n+1} - y_n = h \sum_{j=0}^{k} \gamma_j^* \nabla^j f_{n+1}, \quad \gamma_0^* = 1, \quad \gamma_j^* = (-1)^j \int_0^1 \binom{s+1}{j} ds. \]

**Example 6.7.** We get
\[ \gamma_0^* = 1, \quad \gamma_1^* = \int_0^1 (s-1)ds = -\frac{1}{2}, \quad \gamma_2^* = \int_0^1 \frac{(s-1)s}{2} ds = -\frac{1}{12} \]
and the first methods becomes
\begin{align*}
y_{n+1} - y_n &= hf_{n+1} \quad \text{(Backward Euler)} \\
y_{n+1} - y_n &= h \left( \frac{1}{2} f_{n+1} + \frac{1}{2} f_n \right) \quad \text{(Trapezoidal method)} \\
y_{n+1} - y_n &= h \left( \frac{5}{12} f_{n+1} + \frac{2}{3} f_n - \frac{1}{12} f_{n-1} \right).
\end{align*}

A $k$-step Adams-Moulton method is implicit, of order $k+1$ and is zero-stable. The error constant $C_{p+1} = \gamma_{k+1}^*$. Despite the fact that the Adams-Moulton methods are implicit, they have some advantages compared to their explicit counterparts: They are of one order higher, the error constants are much smaller, and the linear stability properties (when the methods are applied to the linear test problem $y' = \lambda y$) are much better.

<table>
<thead>
<tr>
<th>$k$</th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\gamma_k$</td>
<td>1</td>
<td>$\frac{1}{2}$</td>
<td>$\frac{5}{12}$</td>
<td>$\frac{3}{8}$</td>
<td>$\frac{251}{720}$</td>
<td>$\frac{95}{288}$</td>
<td>$\frac{19087}{60480}$</td>
</tr>
<tr>
<td>$\gamma_{k+1}^*$</td>
<td>1</td>
<td>$\frac{1}{2}$</td>
<td>$\frac{1}{12}$</td>
<td>$\frac{1}{24}$</td>
<td>$\frac{19}{720}$</td>
<td>$\frac{3}{160}$</td>
<td>$\frac{863}{60480}$</td>
</tr>
</tbody>
</table>

Table 1: The $\gamma$'s for the Adams methods.

### 6.5 Predictor-corrector methods

A predictor-corrector (PC) pair is a pair of one explicit (predictor) and one implicit (corrector) methods. The nonlinear equations from the application of the implicit method are solved by a fixed number of fixed point iterations, using the solution by the explicit method as starting values for the iterations.
Example 6.8. We may construct a PC method from a second order Adams-Bashforth scheme and the trapezoidal rule as follows:

\[ y_{n+1}^{[0]} = y_n + \frac{h}{2}(3f_n - f_{n-1}) \quad (P: \text{Predictor}) \]

for \( l = 0, 1, \ldots, m \)

\[ f_{n+1}^{[l]} = f(t_{n+1}, y_{n+1}^{[l]}) \quad (E: \text{Evaluation}) \]

\[ y_{n+1}^{[l+1]} = y_n + \frac{h}{2}(f_{n+1}^{[l]} + f_n) \quad (C: \text{Corrector}) \]

end

\[ y_{n+1} = y_{n+1}^{[m]} \]

\[ f_{n+1} = f(t_{n+1}, y_{n+1}). \quad (E: \text{Evaluation}) \]

Such schemes are commonly referred as \( P(EC)^m \) E schemes.

The predictor and the corrector is often by the same order, in which case only one or two iterations are needed.

Error estimation in predictor-corrector methods.

The local discretization error of some LMM is given by

\[ h\tau_{n+1} = \sum_{l=0}^{k} (\alpha_l y(t_{n-k+1+l} - h\beta_l y'(t_{n-k+1+l})) = h^{p+1}C_{p+1}y^{(p+1)}(t_{n-k+1}) + O(h^{p+2}). \]

But we can do the Taylor expansions of \( y \) and \( y' \) around \( t_n \) rather than \( t_{n-k+1} \). This will not alter the principal error term, but the terms hidden in the expression \( O(h^{p+2}) \) will change. As a consequence, we get

\[ h\tau_{n+1} = h^{p+1}C_{p+1}y^{(p+1)}(t_n) + O(h^{p+2}). \]

Assume that \( y_i = y(t_i) \) for \( i = n - k + 1, \ldots, n \), and \( \alpha_k = 1 \). Then

\[ h\tau_{n+1} = y(t_{n+1}) - y_{n+1} + O(h^{p+2}) = h^{p+1}C_{p+1}y^{(p+1)}(t_n) + O(h^{p+2}). \]

Assume that we have chosen a predictor-corrector pair, using methods of the same order \( p \). Then

\[ (P) \quad y(t_{n+1}) - y_{n+1}^{[0]} \approx h^{p+1}C_{p+1}^{[0]} y^{(p+1)}(t_n), \]

\[ (C) \quad y(t_{n+1}) - y_{n+1} \approx h^{p+1}C_{p+1} y^{(p+1)}(t_n), \]

and

\[ y_{n+1} - y_{n+1}^{[0]} \approx h^{p+1}(C_{p+1}^{[0]} - C_{p+1}) y^{(p+1)}(t_n). \]
From this we get the following local error estimate for the corrector, called Milne’s device:
\[ y(t_{n+1}) - y_{n+1} \approx \frac{C_{p+1}}{C_{p+1}^{[0]} - C_{p+1}} (y_{n+1} - y_{n+1}^{[0]}). \]

**Example 6.9.** Consider the PC-scheme of Example 6.8. In this case
\[ C_{p+1}^{[0]} = \frac{5}{12}, \quad C_{p+1} = -\frac{1}{12}, \quad \text{so} \quad \frac{C_{p+1}}{C_{p+1}^{[0]} - C_{p+1}} = -\frac{1}{6}. \]

Apply the scheme to the linear test problem
\[ y' = -y, \quad y(0) = 1, \]
using \( y_0 = 1, \ y_1 = e^{-h} \) and \( h = 0.1 \). One step of the PC-method gives

| \( l \) | \( y_2^{[l]} \) | \( |y_2 - y_2^{[l]}| \) | \( |y(0.2) - y_2^{[l]}| \) | \( \frac{1}{5} |y_2^{[l]} - y_2^{[0]}| \) |
|---|---|---|---|---|
| 0 | 0.819112 | 4.49 \times 10^{-4} | 3.81 \times 10^{-4} |
| 1 | 0.818640 | 2.25 \times 10^{-5} | 9.08 \times 10^{-5} | 7.86 \times 10^{-5} |
| 2 | 0.818664 | 1.12 \times 10^{-6} | 6.72 \times 10^{-5} | 7.47 \times 10^{-5} |
| 3 | 0.818662 | 5.62 \times 10^{-8} | 6.84 \times 10^{-5} | 7.49 \times 10^{-5} |

After 1-2 iterations, the iteration error is much smaller than the local error, and we also observe that Milne’s device gives a reasonable approximation to the error.

**Remark** Predictor-corrector methods are not suited for stiff problems. You can see this by e.g. using the trapezoidal rule on \( y' = \lambda y \). The trapezoidal rule has excellent stability properties. But the iteration scheme
\[ y_{n+1}^{[l+1]} = y_n + \frac{h}{2} \lambda (y_{n+1}^{[l]} + y_n) \]
will only converge if \( |h\lambda/2| < 1 \).

For stiff system, the Backward differentiation formulas (BDF) is to be preferred. Those are derived in exercise 5.

**References**
