I. Find the local truncation error of the backward Euler’s method, given by Eq. (1.15)

\[ Y_{n+1} = Y_n + hf(t_{n+1}, Y_{n+1}), \quad n = 0, 1, \ldots, \]

of the textbook by Iserles.

Local Truncation Error (LTE):

\[ LTE := y(t_{n+1}) - [y(t_n) + hf(t_{n+1}, y(t_{n+1}))] \]

By Taylor expansion

\[ = y(t_n + h) - (y(t_n) + hy'(t_n + h)) \]

\[ = y(t_n) + hy'(t_n) + O(h^2) - (y(t_n) + hy'(t_n) + O(h^2)) \]

\[ = O(h^2) \]

Hence, the backward Euler’s method is of order 1.

II. Apply the method of proof of Theorems 1.1 and 1.2 of the textbook by Iserles to prove convergence of the implicit midpoint rule, Eq. (1.12)

\[ Y_{n+1} = Y_n + hf \left( t_n + \frac{h}{2}, \frac{1}{2}(Y_n + Y_{n+1}) \right). \]

in the textbook by Iserles.

Proof:

At first, we must check the order of midpoint rule. We substitute the exact solution to the rule,

\[ y(t_{n+1}) - [y(t_n) + hf(t_{n+1}, y(t_{n+1}))] \]

\[ = \left[ y(t_n) + hy'(t_n) + \frac{1}{2}h^2y''(t_n) + O(h^3) \right] \]

\[ - \left\{ y(t_n) + h\left[ y'(t_n) + \frac{1}{2}hy''(t_n) + O(h^2) \right] \right\} = O(h^3) \]

Therefore the midpoint rule is of order two.

Let \( e_n = y_n - y(t_n) \). We know that

\[ y(t_{n+1}) = y(t_n) + hf \left( t_n + \frac{h}{2}, \frac{1}{2}(y(t_n) + y(t_{n+1})) \right) + O(h^3). \]

We subtract the above equality from midpoint rule, giving

\[ e_{n+1} = e_n + h \left[ f \left( t_n + \frac{h}{2}, \frac{1}{2}(y_n + y_{n+1}) \right) - f \left( t_n + \frac{h}{2}, \frac{1}{2}(y(t_n) + y(t_{n+1})) \right) \right] + O(h^3). \]
Thus, it follows by the triangle inequality from the Lipschitz condition that
\[
\|e_{n+1}\| \leq \|e_n\| + h\left\| f\left(t_n + \frac{h}{2}, \frac{1}{2}(y_n + y_{n+1})\right) - f\left(t_n + \frac{h}{2}, \frac{1}{2}(y(t_n) + y(t_{n+1}))\right)\right\| + ch^3
\]
\[
\leq \|e_n\| + h\lambda \left\| \frac{1}{2} (e_n + e_{n+1}) \right\| + ch^3
\]
\[
\Rightarrow \|e_{n+1}\| \leq \left(1 + \frac{\frac{1}{2}h\lambda}{1 - \frac{1}{2}h\lambda}\right) \|e_n\| + \frac{c}{1 - \frac{1}{2}h\lambda}h^3
\]
Then we claim that
\[
\|e_n\| \leq \frac{c}{\lambda} \left[\left(1 + \frac{h\lambda}{1 - \frac{1}{2}h\lambda}\right)^n - 1\right] h^2, \quad n = 0, 1, \ldots
\]
The proof is by induction on \(n\). When \(n = 0\), it is obvious that \(e_0 \leq 0\). For general \(n \geq 0\) we assume that it is true to \(n\) then we have
\[
\|e_{n+1}\| \leq \left(1 + \frac{\frac{1}{2}h\lambda}{1 - \frac{1}{2}h\lambda}\right) \|e_n\| + \frac{c}{1 - \frac{1}{2}h\lambda}h^3
\]
Then we proves it is true. As
\[
\frac{1 + \frac{\frac{1}{2}h\lambda}{1 - \frac{1}{2}h\lambda}}{1 - \frac{1}{2}h\lambda} = 1 + \frac{h\lambda}{1 - \frac{1}{2}h\lambda} \leq \exp\left(\frac{h\lambda}{1 - \frac{1}{2}h\lambda}\right).
\]
Consequently, it yields
\[
\|e_n\| \leq \frac{ch^2}{\lambda} \left[\left(1 + \frac{\frac{1}{2}h\lambda}{1 - \frac{1}{2}h\lambda}\right)^n - 1\right] \leq \frac{ch^2}{\lambda} \exp\left(\frac{nh\lambda}{1 - \frac{1}{2}h\lambda}\right) \leq \frac{ch^2}{\lambda} \exp\left(\frac{t^*\lambda}{1 - \frac{1}{2}h\lambda}\right) \text{ where } nh \leq t^*.
\]
Therefore
\[
\lim_{h \to 0, 0 \leq nh \leq t^*} \|e_n\| = 0.
\]
In other words, the midpoint rule converges.

IV. Exercise 2.1 of the textbook by Iserles.
2.1 Derive explicitly the three-step and four-step Adams-Moulton methods and the three-step Adams-Bashforth method.

Solution:
The \(b\) coefficients of the AdamsCMoulton methods are chosen to obtain the highest order possible.
And the AdamsCMoulton methods are implicit methods.
Four-step Adams-Moulton method:
\(S = 3\), we take \(t_{n+3} = 1, t_{n+2} = 0, t_{n+1} = -1, t_n = -2\) and \(p_1 = 1, p_2 = t, p_3 = t^2, p_4 = t^3\), then we have
\[
\int_0^1 p_i dt = b_3 p_i(1) + b_2 p_i(0) + b_1 p_i(-1) + b_0 p_i(-2), \quad i = 1, 2, 3, 4.
\]
Then we have

\[
\begin{align*}
  b_3 + b_2 + b_1 + b_0 &= 1 \\
  b_3 - b_1 - 2b_0 &= \frac{1}{2} \\
  b_3 + b_1 + 4b_0 &= \frac{1}{3} \\
  b_3 - b_1 - 8b_0 &= \frac{1}{4}
\end{align*}
\Rightarrow \begin{align*}
  b_3 &= \frac{3}{8} \\
  b_2 &= \frac{19}{24} \\
  b_1 &= -\frac{5}{24} \\
  b_0 &= \frac{1}{24}
\end{align*}
\]

Thus we have the three-step Adams-Moulton method:

\[
y_{n+3} = y_{n+2} + h \left[ \frac{3}{8} f(t_{n+3}, y_{n+3}) + \frac{19}{24} f(t_{n+2}, y_{n+2}) - \frac{5}{24} f(t_{n+1}, y_{n+1}) + \frac{1}{24} f(t_n, y_n) \right].
\]

Four-step Adams-Moulton method:

\( S = 4 \), we take \( t_n = 1, t_{n+1} = 0, t_{n+2} = -1, t_{n+3} = -2, t_{n+4} = -3 \) and \( p_1 = 1, p_2 = t, p_3 = t^2, p_4 = t^3, p_5 = t^4 \), then we have

\[
\int_0^1 p_i dt = b_4 p_1(1) + b_3 p_i(0) + b_2 p_i(-1) + b_1 p_i(-2) + b_0 p_i(-3), i = 1, \ldots, 5.
\]

Then we have

\[
\begin{align*}
  b_4 + b_3 + b_2 + b_1 + b_0 &= 1 \\
  b_4 - b_2 - 2b_1 - 3b_0 &= \frac{1}{2} \\
  b_4 + b_2 + 4b_1 + 9b_0 &= \frac{1}{3} \\
  b_4 - b_2 - 8b_1 - 27b_0 &= \frac{1}{4} \\
  b_4 - b_2 + 16b_1 + 81b_0 &= \frac{1}{5}
\end{align*}
\Rightarrow \begin{align*}
  b_4 &= \frac{251}{720} \\
  b_3 &= \frac{323}{360} \\
  b_2 &= -\frac{11}{30} \\
  b_1 &= \frac{53}{360} \\
  b_0 &= -\frac{19}{720}
\end{align*}
\]

Thus we have the three-step Adams-Moulton method:

\[
y_{n+4} = y_{n+3} + h \left[ \frac{251}{720} f(t_{n+4}, y_{n+4}) + \frac{323}{360} f(t_{n+3}, y_{n+3}) - \frac{11}{30} f(t_{n+2}, y_{n+2}) + \frac{53}{360} f(t_{n+1}, y_{n+1}) - \frac{19}{720} f(t_n, y_n) \right].
\]

Three-step Adams-Bashforth method:

\( S = 3 \), then we have

\[
\begin{align*}
  l_0 &= \frac{(t - t_{n+1})(t - t_{n+2})}{(t_n - t_{n+1})(t_n - t_{n+2})} = \frac{1}{2h^2}(t - t_{n+1})(t - t_{n+2}) \\
  l_1 &= \frac{(t - t_{n})(t - t_{n+2})}{(t_{n+1} - t_n)(t_{n+1} - t_{n+2})} = \frac{1}{h^2}(t - t_n)(t - t_{n+2}) \\
  l_2 &= \frac{(t - t_{n})(t - t_{n+1})}{(t_{n+2} - t_n)(t_{n+2} - t_{n+1})} = \frac{1}{2h^2}(t - t_n)(t - t_{n+1})
\end{align*}
\]
Next, we derive \( b \)

\[
b_0 = \frac{1}{h} \int_{t_{n+2}}^{t_{n+3}} l_0(\tau) d\tau = \frac{1}{2h^3} \int_{t_{n+2}}^{t_{n+3}} (\tau - t_{n+1})(\tau - t_{n+2}) d\tau
\]

\[
= \frac{1}{2h^3} \int_{0}^{h} (\tau + h) \tau d\tau = \frac{1}{2h^3} \left( \frac{\tau^3}{3} + \frac{h\tau^2}{2} \right)_{0}^{h} = \frac{5}{12}.
\]

\[
b_1 = \frac{1}{h} \int_{t_{n+2}}^{t_{n+3}} l_1(\tau) d\tau = -\frac{1}{h^3} \int_{t_{n+2}}^{t_{n+3}} (\tau - t_n)(\tau - t_{n+1}) d\tau
\]

\[
= -\frac{1}{h^3} \int_{0}^{h} (\tau + 2h) \tau d\tau = \frac{1}{h^3} \left( \frac{\tau^3}{3} + h\tau^2 \right)_{0}^{h} = -\frac{4}{3}.
\]

\[
b_2 = \frac{1}{h} \int_{t_{n+2}}^{t_{n+3}} l_2(\tau) d\tau = \frac{1}{2h^3} \int_{t_{n+2}}^{t_{n+3}} (\tau - t_n)(\tau - t_{n+1}) d\tau
\]

\[
= \frac{1}{2h^3} \int_{0}^{h} (\tau + 2h)(\tau + h) d\tau
\]

\[
= \frac{1}{2h^3} \left( \frac{\tau^3}{3} + 3h\tau^2 + 2h^2\tau \right)_{0}^{h} = \frac{23}{12}.
\]

Hence,

\[
y_{n+3} = y_{n+2} + h \left[ \frac{23}{12} f(t_{n+2}, y_{n+2}) - \frac{4}{3} f(t_{n+1}, y_{n+1}) + \frac{5}{12} f(t_n, y_n) \right].
\]

V. Computer Project: Consider the ODE

\[ y' = \frac{1}{t^2} - \frac{y}{t} - y^2, \]

with the initial conditions \( y(1) = -1 \).

Solve the IVP of the ODE for \( t \) in \([1, 5]\), using the trapezoidal rule and Newton’s method for solving the nonlinear equation. Could you verify numerically that the order of the method is two?

Solution:

We want to use Newton method to solve the nonlinear equation. Then we have

\[
f(y_{n+1}) = y_n + \frac{1}{2h} \left( \frac{1}{t^2_n} - \frac{y_n}{t_n} - y_n^2 + \frac{1}{t^2_{n+1}} - \frac{y_{n+1}}{t_{n+1}} - y_{n+1}^2 \right) - y_{n+1},
\]

\[
f'(y_{n+1}) = -\frac{h}{2t_{n+1}} - hy_{n+1} - 1.
\]

Also, use the above results, we can have the order of trapezoid rule is 2 by equation.

\[
p = \log \left( \frac{\|y_{n+1}(h_1) - y_{n+1}(\frac{h_1}{2})\|}{\|y_{n+1}(\frac{h_2}{2}) - y_{n+1}(\frac{h_2}{4})\|} \right) / \log \left( \frac{h_1}{h_2} \right).
\]
Appendix

code for computer problem)
clear all;
clc;

% initial process;
tstart = 1;
tend = 5;
h = [0.1 0.05 0.01];
ytstar = zeros(1, 3);
for j = 1:3;
    n = (tend - tstart) / h(j);
tmp = [tstart, h(j) * ones(1, n)];
t = cumsum(tmp);
y = zeros(1, n + 1);
y(1) = -1;
    for i = 1:n;
        y(i + 1) = newton(h(j), t(i), t(i + 1), y(i));
    end;

figure
plot(t, y, '-')
grid on;
xlabel('t');
ylabel('y');
title(strcat('Solution by Trapzoid Rule when stepsiz=', num2str(h(j))));
hold on
ytstar(j) = y(n + 1);
end;

p = log(abs(ytstar(1) - ytstar(3)) / abs(ytstar(2) - ytstar(3))) / log(h(1) / h(2));
disp(strcat('The order of Trapzoid Rule is ', num2str(p)));

function [y2] = newton(h, t1, t2, y1)
% define initial point.
x0 = 0;
N=100;
x=x0;

% define tolerance;
tau=1e-9;

for i=1:N;
    J=-(h^2*x-h/t2/2-1);
    f=y1+1/2*h*(1/t1^2-y1/t1-y1^2+1/t2^2-x/t2-x^2)-x;
    step=-f/J;
    xs=x+step;
    if norm(xs-x)<tau;
        break;
    else
        x=xs;
    end;
end;
y2=xs;