CPSC 403/542
Assignment 2 - Solutions

1. (a)

$$\begin{aligned} K_i &= f(t_i, Y_i) \\ t_i &= t_{n-1} + h_n c_i \end{aligned}$$

   (b) This proof is literally the same as the one given in the notes following Theorem 3.1, where in place of f you substitute ψ.

2. Around 10000 steps (more than 8000 anyway) are needed.

3. (a) Assuming $f = f(y)$ for notational simplicity,

$$\begin{aligned} y_n^{2,1} &= [y_{n-1} + h/2 f(y_{n-1})] + h/2 f(y_{n-1} + h/2 f(y_{n-1})) \\ &= y_{n-1} + h/2[f(y_{n-1}) + f(y_{n-1} + h/2 f(y_{n-1}))]. \end{aligned}$$

   Multiplying this by 2 and subtracting $y_n^{1,1} = y_{n-1} + hf(y_{n-1})$ then yields

$$y_n^{2,2} = y_{n-1} + hf(y_{n-1} + h/2 f(y_{n-1})).$$

   This is precisely the explicit midpoint method.

   (b) From the error expression for $y_n^{k,1}$, defining $y_n^{k,2} = k y_n^{k,1} - (k-1) y_n^{k-1,1}$ yields the error

$$y(t_n) - y_n^{k,2} = \sum_{j \geq 2} (k^{1-j} - (k-1)^{1-j}) C_j h^j.$$

   Substituting into this $j = 2$ and the two values $k = 2, 3$, the coefficients of the leading terms in the errors for $y_n^{2,2}$ and $y_n^{3,2}$ are $-\frac{1}{2} C_2 h^2$ and $-\frac{1}{6} C_3 h^3$, respectively. Thus,

$$y_n^{3,3} = \frac{1}{2}(3 y_n^{3,2} - y_n^{2,2})$$

   has the $2nd$ order error term eliminated, resulting in a third order method.

   (c) We know already that $y_n^{1,1}$ is forward Euler and $y_n^{2,2}$ is explicit midpoint. They require 1 and 2 function evaluations, respectively. Now consider $y_n^{3,3}$. Let

$$\begin{aligned} Y_1 &= y_{n-1}, \\ Y_{1,2} &= y_{n-1} + \frac{h}{2} f(Y_1), \end{aligned}$$

$$
\begin{aligned}
y_n^{2,1} &= y_{n-1} + \frac{h}{2}(f(Y_1) + f(Y_{1,2})), \\
Y_{1,3} &= y_{n-1} + \frac{h}{3} f(Y_1), \\
Y_{2,3} &= Y_{1,3} + \frac{h}{3} f(Y_{1,3}) = y_{n-1} + \frac{h}{3}(f(Y_1) + f(Y_{1,3})), \\
y_n^{3,1} &= Y_{2,3} + \frac{h}{3} f(Y_{2,3}) = y_{n-1} + \frac{h}{3}(f(Y_1) + f(Y_{1,3}) + f(Y_{2,3})).
\end{aligned}
$$

Then

$$
\begin{aligned}
y_n^{2,2} &= y_{n-1} + h f(Y_{1,2}), \\
y_n^{3,2} &= y_{n-1} + h(f(Y_{1,3}) + f(Y_{2,3}) - f(Y_{1,2})), \\
y_n^{3,3} &= y_{n-1} + \frac{h}{2}(3f(Y_{1,3}) + 3f(Y_{2,3}) - 4f(Y_{1,2})).
\end{aligned}
$$

In tableau form we can write

$$
\begin{array}{c|cccc}
0 & 0 & 0 & 0 & 0 \\
\frac{1}{2} & \frac{1}{2} & 0 & 0 & 0 \\
\frac{1}{3} & \frac{1}{3} & 0 & 0 & 0 \\
\frac{2}{3} & \frac{1}{3} & 0 & \frac{1}{3} & 0 \\
\hline
 & 1 & 0 & 0 & 0 \\
\hline
 & 0 & 1 & 0 & 0 \\
\hline
 & 0 & -2 & \frac{3}{2} & \frac{3}{2}
\end{array}
$$

The last three lines are different quadrature weights b_i corresponding to the same coefficient matrix A. Of course, if $b_i = 0, \ \forall i > j$ then there is no reason to consider anything but the upper left $j \times j$ block of A.

From this description it is obvious that the 3rd order method costs 4 function evaluations. Even though it forms an embedded pair with $y_n^{2,2}$, there are better embedded pairs where the 3rd order method costs only 3 function evaluations.

(d) The following table records maximum errors and convergence rates at $t = 25$. The claimed orders are clearly demonstrated.

h	forward Euler	rate	$y_n^{2,2}$	rate	$y_n^{3,3}$	rate
.02	1.29e-7		7.23e-9		3.20e-10	
.01	6.48e-8	1.00	1.71e-9	2.08	3.81e-11	3.07
.005	3.24e-8	1.00	4.17e-10	2.08	4.63e-12	3.04

(e) It is clear from the above that in general, k steps of an s-stage Runge-Kutta method can be written as one Runge-Kutta method with ks stages. The linear combination of such methods, which is what extrapolation does, yields another Runge-Kutta method, in general using at most $\sum_{j=1}^{k} sj = sk(k+1)/2$ stages. Applying extrapolation based on Euler's method, we obtain for an arbitrary positive integer k that $y_n^{k,k}$ is a Runge-Kutta method of order k using at most $\frac{k(k+1)}{2}$ stages.

4. (a)

$$
\begin{aligned}
y' &= \lambda(y - g(t)) \\
h^{-1}\sum_{j=0}^{k}\alpha_j y_{n-1} &= \lambda\sum_{j=0}^{k}\beta_j(y_{n-j} - g(t_{n-j}))
\end{aligned}
$$

Divide by λ and let $\lambda h \to -\infty$. Then

$$0 = \sum_{j=0}^{k}\beta_j(y_{n-j} - g(t_{n-j}))$$

We must have $\beta_0 \neq 0$ because otherwise y_n is arbitrary. Then

$$y_n - g(t_n) = \beta_0^{-1}\sum_{j=1}^{k}\beta_j(y_{n-j} - g(t_{n-j}))$$

Since g is arbitrary, the only way to make the sum on the right hand side always equal 0 is to require $\beta_j = 0, \forall j > 0$.

This leaves the coefficients $\beta_0, \alpha_1, \ldots, \alpha_k$. These are $k+1$ coefficients and to obtain a method of order k they must satisfy $k+1$ linearly independent order conditions, as in the text. This gives the BDF schemes uniquely.

(b) We have just shown that the only 7-step method of order 7 which has stiff decay is BDF, and we know that all BDF methods with more than 6 steps are not strongly stable.

However, it is possible to consider an 8-step method of order 7. We set $\beta_j = 0,\ j > 0$, which is the condition for stiff decay. This gives 9 coefficient $\beta_0, \alpha_1, \ldots, \alpha_8$ to satisfy 8 order conditions. So we are left with a one-parameter family of methods. There are a number of methods in this family which are strongly stable. One such method, which Doug has found, is given by

β_0	α_1	α_2	α_3	α_4
0.400208	−2.5	3.147089	−2.558905	0.864085
	α_5	α_6	α_7	α_8
	0.429314	−0.588184	0.244283	−0.037682