Department of Mathematics, University of Utah

Analysis of Numerical Methods, II

MATH 6620 – Section 001 – Spring 2024

Homework 3 Solutions

Time-stepping methods, II

Due Friday, February 23, 2024

Submit your solutions online through Gradescope.

1. (Runga-Kutta Methods)

a. Recall Ralston's method from the previous assignment:

$$\boldsymbol{u}_{n+1} = \boldsymbol{u}_n + \frac{k}{4}\boldsymbol{f}(t_n, \boldsymbol{u}_n) + \frac{3k}{4}\boldsymbol{f}\left(t_n + \frac{2}{3}k, \boldsymbol{u}_n + \frac{2}{3}k\boldsymbol{f}(t_n, \boldsymbol{u}_n)\right),$$

Identify the Butcher tableau for this method.

b. Show that Ralston's method is consistent to second order.

<u>**Solution**</u>:

a. We rewrite Ralston's method as the following two-stage approach:

$$\begin{aligned}
\boldsymbol{U}_1 &= \boldsymbol{u}_n + \boldsymbol{0} \\
\boldsymbol{U}_2 &= \boldsymbol{u}_n + k\frac{2}{3}\boldsymbol{f}(t_n, \boldsymbol{U}_1) \\
\boldsymbol{u}_{n+1} &= \boldsymbol{u}_n + \frac{k}{4}\boldsymbol{f}(t_n + 0k, \boldsymbol{U}_1) + \frac{3k}{4}\boldsymbol{f}(t_n + 2k/3, \boldsymbol{U}_2).
\end{aligned}$$

From this form, we can immediately read off the Butcher tableau coefficients:

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

b. We proceed to compute the LTE for this scheme, which we write as,

$$\begin{aligned}
\text{LTE} &= \frac{\boldsymbol{u}(t_{n+1}) - \boldsymbol{u}(t_n)}{k} - \frac{1}{4}\boldsymbol{f}(t_n, \boldsymbol{u}(t_n)) - \frac{3}{4}\boldsymbol{f}(t_n + 2k/3, \boldsymbol{U}_2) \\
&= \frac{\boldsymbol{u}(t_{n+1}) - \boldsymbol{u}(t_n)}{k} - \frac{1}{4}\boldsymbol{u}'(t_n) - \frac{3}{4}\boldsymbol{f}(t_n + 2k/3, \boldsymbol{U}_2) \qquad (1)
\end{aligned}$$

Armed with the two Taylor expansions around $t = t_n$ and $(t, \boldsymbol{u}) = (t_n, \boldsymbol{u}(t_n))$, respectively, we have,

$$\begin{aligned}
\frac{\boldsymbol{u}(t_{n+1}) - \boldsymbol{u}(t_n)}{k} &= \boldsymbol{u}'(t_n) + \frac{k}{2}\boldsymbol{u}''(t_n) + \mathcal{O}(k^2) \\
\boldsymbol{f}(t_n + 2k/3, \boldsymbol{U}_2) &= \boldsymbol{f}(t_n, \boldsymbol{u}_n) + k\frac{2}{3}\frac{\partial \boldsymbol{f}}{\partial t}(t_n, \boldsymbol{u}(t_n)) + \frac{\partial \boldsymbol{f}}{\partial \boldsymbol{u}}(t_n, \boldsymbol{u}(t_n))(\boldsymbol{U}_2 - \boldsymbol{u}_1) + \mathcal{O}(k^2) \\
&= \boldsymbol{u}'(t_n) + k\frac{2}{3}\frac{\partial \boldsymbol{f}}{\partial t}(t_n, \boldsymbol{u}(t_n)) + \frac{\partial \boldsymbol{f}}{\partial \boldsymbol{u}}(t_n, \boldsymbol{u}(t_n))(\boldsymbol{U}_2 - \boldsymbol{u}_1) + \mathcal{O}(k^2)
\end{aligned}$$

where we have slightly abused notation, defining $\boldsymbol{U}_2 \coloneqq \boldsymbol{u}(t_n) + 2k/3\boldsymbol{f}(t_n, \boldsymbol{u}(t_n))$, and we have used $\mathcal{O}(\|\boldsymbol{U}_2 - \boldsymbol{u}(t_n)\|^2) = \mathcal{O}(k^2)$. Using these Taylor expansions in (1), we have,

$$\begin{aligned}
\text{LTE} &= \boldsymbol{u}'(t_n) + \frac{k}{2}\boldsymbol{u}''(t_n) - \frac{1}{4}\boldsymbol{u}'(t_n) - \frac{3}{4}\boldsymbol{u}'(t_n) - \frac{3}{4}k\frac{2}{3}\frac{\partial \boldsymbol{f}}{\partial t}(t_n, \boldsymbol{u}(t_n)) \\
&\quad - \frac{3}{4}\frac{2k}{3}\frac{\partial \boldsymbol{f}}{\partial \boldsymbol{u}}(t_n, \boldsymbol{u}(t_n))\boldsymbol{f}(t_n, \boldsymbol{u}(t_n)) + \mathcal{O}(k^2) \\
&= \frac{k}{2}\boldsymbol{u}''(t_n) - \frac{k}{2}\left(\frac{\partial \boldsymbol{f}}{\partial t}(t_n, \boldsymbol{u}(t_n)) + \frac{\partial \boldsymbol{f}}{\partial \boldsymbol{u}}(t_n, \boldsymbol{u}(t_n))\boldsymbol{u}'(t_n)\right) + \mathcal{O}(k^2) \\
&\overset{(*)}{=} \frac{k}{2}\boldsymbol{u}''(t_n) - \frac{k}{2}\boldsymbol{u}''(t_n) + \mathcal{O}(k^2) = \mathcal{O}(k^2).
\end{aligned}$$

where $(*)$ uses the chain rule, $\frac{\mathrm{d}}{\mathrm{d}t}\boldsymbol{u}'(t_n) = \frac{\mathrm{d}}{\mathrm{d}t}\boldsymbol{f}(t_n, \boldsymbol{u}(t_n)) = \frac{\partial \boldsymbol{f}}{\partial t}(t_n, \boldsymbol{u}(t_n)) + \frac{\partial \boldsymbol{f}}{\partial \boldsymbol{u}}(t_n, \boldsymbol{u}(t_n))\boldsymbol{u}'(t)$.

2. (Multi-step methods)

a. Compute coefficients for the following implicit multi-step scheme that achieves the optimal order of accuracy,

$$\boldsymbol{u}_{n+1} + \alpha_1\boldsymbol{u}_n + \alpha_2\boldsymbol{u}_{n-1} = k\beta_0\boldsymbol{f}_{n+1} + k\beta_1\boldsymbol{f}_n + k\beta_2\boldsymbol{f}_{n-1},$$

where $\boldsymbol{f}_j \coloneqq \boldsymbol{f}(t_j, \boldsymbol{u}_j)$.

b. Identify the order of consistency of the scheme, and determine whether this method is 0-stable and/or A-stable.

__**Solution**__:

a. The LTE for this scheme reads,

$$\begin{aligned}
\text{LTE} &= \frac{1}{k}\boldsymbol{u}(t_{n+1}) + \frac{\alpha_1}{k}\boldsymbol{u}(t_n) + \frac{\alpha_2}{k}\boldsymbol{u}(t_{n-1}) - \beta_0\boldsymbol{f}(t_{n+1}, \boldsymbol{u}(t_{n+1})) - \beta_1\boldsymbol{f}(t_n, \boldsymbol{u}(t_n)) - \beta_2\boldsymbol{f}(t_{n-1}, \boldsymbol{u}(t_{n-1})) \\
&= \frac{1}{k}\boldsymbol{u}(t_{n+1}) + \frac{\alpha_1}{k}\boldsymbol{u}(t_n) + \frac{\alpha_2}{k}\boldsymbol{u}(t_{n-1}) - \beta_0\boldsymbol{u}'(t_{n+1}) - \beta_1\boldsymbol{u}'(t_n) - \beta_2\boldsymbol{u}'(t_{n-1})
\end{aligned}$$

With the abbreviations $\boldsymbol{u} = \boldsymbol{u}(t_{n-1})$, $\boldsymbol{u}' = \boldsymbol{u}'(t_{n-1})$, etc., we employ the following Taylor series expansions:

$$\begin{aligned}
\boldsymbol{u}(t_{n+1}) &= \boldsymbol{u} + 2k\boldsymbol{u}' + 2k^2\boldsymbol{u}'' + \frac{4k^3}{3}\boldsymbol{u}''' + \frac{2k^4}{3}\boldsymbol{u}'''' + \ldots \\
\boldsymbol{u}(t_n) &= \boldsymbol{u} + k\boldsymbol{u}' + \frac{k^2}{2}\boldsymbol{u}'' + \frac{k^3}{6}\boldsymbol{u}''' + \frac{k^4}{24}\boldsymbol{u}'''' + \ldots \\
\boldsymbol{u}'(t_{n+1}) &= \boldsymbol{u}' + 2k\boldsymbol{u}'' + 2k^2\boldsymbol{u}''' + \frac{4k^3}{6}\boldsymbol{u}'''' + \ldots \\
\boldsymbol{u}'(t_n) &= \boldsymbol{u}' + k\boldsymbol{u}'' + \frac{k^2}{2}\boldsymbol{u}''' + \frac{k^3}{6}\boldsymbol{u}'''' + \ldots
\end{aligned}$$

Using these in the LTE expression and collecting terms with the same order in k, we obtain the following system of linear equations:

$$\begin{aligned}
\mathcal{O}(1/k):& \qquad & 1+\alpha_1+\alpha_2 &= 0\\
\mathcal{O}(1):& & 2+\alpha_1-\beta_0-\beta_1-\beta_2 &= 0\\
\mathcal{O}(k):& & 2+\frac{\alpha_1}{2}-2\beta_0-\beta_1 &= 0\\
\mathcal{O}(k^2):& & \frac{4}{3}+\frac{\alpha_1}{6}-2\beta_0-\frac{\beta_1}{2} &= 0\\
\mathcal{O}(k^3):& & \frac{2}{3}+\frac{\alpha_1}{24}-\frac{4\beta_0}{3}\beta_0-\frac{\beta_1}{6} &= 0,
\end{aligned}$$

i.e.,

$$\begin{pmatrix} 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & -1 & -1 & -1 \\ \frac{1}{2} & 0 & -2 & -1 & 0 \\ \frac{1}{6} & 0 & -2 & -\frac{1}{2} & 0 \\ \frac{1}{24} & 0 & -\frac{4}{3} & -\frac{1}{6} & 0 \end{pmatrix} \begin{pmatrix} \alpha_1 \\ \alpha_2 \\ \beta_0 \\ \beta_1 \\ \beta_2 \end{pmatrix} = \begin{pmatrix} -1 \\ -2 \\ -2 \\ -\frac{4}{3} \\ -\frac{2}{3} \end{pmatrix},$$

whose solution is,

$$(\alpha_1,\alpha_2,\beta_0,\beta_1,\beta_2) = \left(0,-1,\frac{1}{3},\frac{4}{3},\frac{1}{3}\right).$$

Thus, the scheme is,

$$\boldsymbol{u}_{n+1}-\boldsymbol{u}_{n-1} = \frac{k}{3}\boldsymbol{f}_{n+1}+\frac{4k}{3}\boldsymbol{f}_n+\frac{k}{3}\boldsymbol{f}_{n-1}.$$

b. From the previous part, all terms up to order $\mathcal{O}(k^4)$ were eliminated. Hence, this scheme is accurate to fourth order. To investigate stability, we identify the characteristic polynomials from the coefficients determined in the previous part:

$$\begin{aligned}
\rho(w) &:= w^2+\alpha_1 w+\alpha_2 = w^2-1\\
\sigma(w) &:= \beta_0 w^2+\beta_1 w+\beta_2 = \frac{w^2}{3}+\frac{4w}{3}+\frac{1}{3}.
\end{aligned}$$

The roots of $\rho(w)$ are $w=\pm 1$, which are simple roots on the unit circle. Hence, ρ satisfies the root condition, and so the scheme is 0-stable. To determine A-stability, it is enough to quote the second Dahlquist barrier: no A-stable multistep method of order greater than 2 exists. Since our scheme is 4th order, it cannot possibly be A stable. However, here is a more formal way to conclude this: To investigate A-stability, we would need the w-polynomial

$$\rho(w)-z\sigma(w)$$

to satisfy the root condition for every $z\in\mathbb{C}$ in the left half-plane. To see if this is plausible, consider a real-valued $z<0$. Then,

$$\rho(w)-z\sigma(w) = w^2(1-z/3)+w(-4z/3)+(z/3-1) \stackrel{\eta:=z/3}{=} w^2(1-\eta)-4\eta w+(\eta-1).$$

The roots of this polynomial are the same as the roots of,

$$w^2 - \frac{4\eta}{1-\eta} w - 1,$$

which are,

$$w = \frac{2\eta}{1-\eta} \pm \sqrt{1 + \left(\frac{2\eta}{1-\eta}\right)^2}.$$

However, since η is negative, then the "$-$" root choice above puts this root outside the unit circle for any $\eta < 0$. Hence, this scheme is not A-stable.

3. (SSP Methods)
In this problem, consider an autonomous ODE, $\boldsymbol{u}' = \boldsymbol{f}(\boldsymbol{u})$.

a. Consider an s-stage explicit Runge-Kutta method. For each $m = 2, \ldots, s+1$, let constants $\{\alpha_{m,j}\}_{j=1}^{m-1}$ be given such that $\alpha_{m,j} \geq 0$ and $\sum_{j=1}^{m-1} \alpha_{m,j} = 1$. Show that such an s-stage explicit method can be written as,

$$\begin{aligned} \boldsymbol{U}_1 &:= \boldsymbol{u}_n, \\ \boldsymbol{U}_m &:= \sum_{j=1}^{m-1} \left(\alpha_{m,j} \boldsymbol{U}_j + \beta_{m,j} \boldsymbol{f}(\boldsymbol{U}_j)\right) & 2 \leq m \leq s+1 \\ \boldsymbol{u}_{n+1} &= \boldsymbol{U}_{s+1} \end{aligned}$$

b. Let $|\cdot|$ be any seminorm on vectors $\boldsymbol{u}$, and suppose that there exists a $k_* > 0$ such that for all $\boldsymbol{u}$ and $k \in (0, k_*]$, then $|\boldsymbol{u} + k\boldsymbol{f}(\boldsymbol{u})| \leq |\boldsymbol{u}|$. Assume that the $\alpha_{m,j}$ coefficients above can be chosen so that $\beta_{m,j} \geq 0$ for all j, m. Show that there is a $c > 0$ such that

$$k \in (0, ck_*] \implies |\boldsymbol{u}_{n+1}| \leq |\boldsymbol{u}_n|,$$

and explicitly identify a formula for c in terms of the $\alpha_{m,j}$ and $\beta_{m,j}$. Schemes that satisfy this are called (Runge-Kutta) *Strong Stability Preserving* (SSP) schemes. The constant c is called the *SSP coefficient*. (The point here is that it's somewhat easy to establish boundedness of the seminorm $|\cdot|$ for a simple Forward Euler scheme; SSP methods allow one to directly port this boundedness to higher order methods.)

c. Verify that the following is an SSP scheme:

$$\begin{array}{c|ccc} 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ \frac{1}{2} & \frac{1}{4} & \frac{1}{4} & 0 \\ \hline & \frac{1}{6} & \frac{1}{6} & \frac{2}{3} \end{array}$$

d. Is the Ralston method from problem 1 an SSP scheme? If so, compute its SSP coefficient.

<u>**Solution**</u>:

a. An s-stage explicit Runge-Kutta method for an autonomous ODE evolving from time $t = t_n$ with stepsize $k \geq 0$ can be written as,

$$\begin{aligned} \boldsymbol{U}_m &= \boldsymbol{u}_n + k \sum_{j=1}^{m-1} a_{m,j} \boldsymbol{f}(\boldsymbol{U}_j), & m \in [s] \\ \boldsymbol{u}_{n+1} &= \boldsymbol{u}_n + k \sum_{j=1}^{s} b_j \boldsymbol{f}(\boldsymbol{U}_j), \end{aligned}$$

where $\{a_{m,j}, b_j\}$ are the Butcher tableau coefficients of the method. To understand the basic idea of the desired transformation, consider $m = 1, 2$:

$$\begin{aligned}
\boldsymbol{U}_1 &= \boldsymbol{u}_n, \\
\boldsymbol{U}_2 &= \boldsymbol{u}_n + k a_{2,1} \boldsymbol{f}(\boldsymbol{U}_1) = \alpha_{2,1} \boldsymbol{U}_1 + \beta_{2,1} \boldsymbol{f}(\boldsymbol{U}_1),
\end{aligned}$$

where on the second line we have used $\boldsymbol{U}_1 = \boldsymbol{u}_n$ and made the assignment:

$$\alpha_{2,1} = 1, \qquad\qquad \beta_{2,1} = k a_{2,1}.$$

In particular, the first equality defining $\boldsymbol{U}_j$ for $j = 1, 2$ above implies,

$$\begin{aligned}
\boldsymbol{u}_n &= \boldsymbol{U}_1 \\
\boldsymbol{u}_n &= \boldsymbol{U}_2 - k a_{2,1} \boldsymbol{f}(\boldsymbol{U}_1).
\end{aligned}$$

Then for $m = 3$, we have:

$$\begin{aligned}
\boldsymbol{U}_3 &= \boldsymbol{u}_n + k a_{3,1} \boldsymbol{f}(\boldsymbol{U}_1) + k a_{3,2} \boldsymbol{f}(\boldsymbol{U}_2) \\
&= \alpha_{3,1} \boldsymbol{u}_n + \alpha_{3,2} \boldsymbol{u}_n + k a_{3,1} \boldsymbol{f}(\boldsymbol{U}_1) + k a_{3,2} \boldsymbol{f}(\boldsymbol{U}_2) \\
&= \alpha_{3,1} \boldsymbol{U}_1 + \alpha_{3,2} \left(\boldsymbol{U}_2 - k a_{2,1} \boldsymbol{f}(\boldsymbol{U}_1) \right) + k a_{3,1} \boldsymbol{f}(\boldsymbol{U}_1) + k a_{3,2} \boldsymbol{f}(\boldsymbol{U}_2) \\
&= \left(\alpha_{3,1} \boldsymbol{U}_1 + \alpha_{3,2} \boldsymbol{U}_2 \right) + \left(\beta_{3,1} \boldsymbol{f}(\boldsymbol{U}_1) + \beta_{3,2} \boldsymbol{f}(\boldsymbol{U}_2) \right),
\end{aligned}$$

where

$$\beta_{3,1} = k(a_{3,1} - \alpha_{3,2} a_{2,1}), \qquad\qquad \beta_{3,2} = k a_{3,2}.$$

Thus, to show that we can do this for arbitrarily large s, we know from the fact that these are explicit Runge-Kutta methods:

$$\boldsymbol{u}_n = \boldsymbol{U}_m - k \sum_{j=1}^{m-1} a_{m,j} \boldsymbol{f}(\boldsymbol{U}_j), \qquad\qquad m \in [s],$$

and therefore for any $m \in [s]$:

$$\begin{aligned}
\boldsymbol{U}_m &= \boldsymbol{u}_n + k \sum_{j=1}^{m-1} a_{m,j} \boldsymbol{f}(\boldsymbol{U}_j) \\
&= \sum_{j=1}^{m-1} \left(\alpha_{m,j} \boldsymbol{u}_n + k a_{m,j} \boldsymbol{f}(\boldsymbol{U}_j) \right) \\
&= \sum_{j=1}^{m-1} \left(\alpha_{m,j} \left(\boldsymbol{U}_j - k \sum_{\ell=1}^{j-1} a_{j,\ell} \boldsymbol{f}(\boldsymbol{U}_\ell) \right) + k a_{m,j} \boldsymbol{f}(\boldsymbol{U}_j) \right), \\
&= \sum_{j=1}^{m-1} \alpha_{m,j} \boldsymbol{U}_j + \beta_{m,j} \boldsymbol{f}(\boldsymbol{U}_j),
\end{aligned}$$

where

$$\beta_{m,j} = k \left(a_{m,j} - \sum_{q=j+1}^{m-1} \alpha_{m,q} a_{q,j} \right), \qquad\qquad m \in [s]$$

This shows the result as desired for $m \leq s$. To show the result for $m = s+1$ is a similar computation:

$$
\begin{aligned}
\boldsymbol{U}_{s+1} &= \boldsymbol{u}_n + k \sum_{j=1}^{s} b_j \boldsymbol{f}(\boldsymbol{U}_j) \\
&= \sum_{j=1}^{s} \left(\alpha_{s+1,j} \boldsymbol{u}_n + k b_j \boldsymbol{f}(\boldsymbol{U}_j)\right) \\
&= \cdots \\
&= \sum_{j=1}^{s} \alpha_{s+1,j} \boldsymbol{U}_j + \beta_{s+1,j} \boldsymbol{f}(\boldsymbol{U}_j),
\end{aligned}
$$

with,

$$
\beta_{s+1,j} = k \left(b_j - \sum_{q=j+1}^{s} \alpha_{m,q} a_{q,j} \right),
$$

b. Semi-norms satisfy the triangle inequality:

$$
|\boldsymbol{u} + \boldsymbol{v}| \leq |\boldsymbol{u}| + |\boldsymbol{v}|.
$$

Using what we have learned from the previous part, then for any $m \in [s+1]$:

$$
\begin{aligned}
\boldsymbol{U}_m &= \sum_{j=1}^{m-1} \left(\alpha_{m,j} \boldsymbol{U}_j + \beta_{m,j} \boldsymbol{f}(\boldsymbol{U}_j)\right) \\
&= \sum_{j=1}^{m-1} \alpha_{m,j} \left[\boldsymbol{U}_j + \frac{\beta_{m,j}}{\alpha_{m,j}} \boldsymbol{f}(\boldsymbol{U}_j) \right] \qquad (2)
\end{aligned}
$$

To make the strategy moving forward very explicit, consider the following "Forward Euler" operator:

$$
\mathrm{FE}(\boldsymbol{u}, k) \coloneqq \boldsymbol{u} + k \boldsymbol{f}(\boldsymbol{u}).
$$

Applying the FE notation to (2), we have,

$$
\boldsymbol{U}_m = \sum_{j=1}^{m-1} \alpha_{m,j} \mathrm{FE}\left(\boldsymbol{U}_j, k_{m,j}\right) \qquad\qquad k_{m,j} = \frac{\beta_{m,j}}{\alpha_{m,j}} \geq 0,
$$

where the inequality on $k_{m,j}$ uses the assumption that $\beta_{m,j} \geq 0$. Pairing this property with the fact that $\alpha_{m,j} \geq 0$ and $\sum_{j=1}^{m-1} \alpha_{m,j} = 1$ shows that, for SSP methods, intermediate RK stages are *convex combinations of forward Euler steps*. Now suppose we choose k such that,

$$
k \leq c k_*, \quad c = \min_{1 \leq j < m \leq s+1} \frac{\alpha_{m,j}}{\beta_{m,j}/k} \implies k_{m,j} \leq k_*.
$$

(Note that $\beta_{m,j}/k$ is independent of k.) Under this choice of k, then $\mathrm{FE}(\boldsymbol{u}, k) \leq |\boldsymbol{u}|$ by assumption. Then by the triangle inequality and the convex weight property of $\{\alpha_{m,j}\}_j$, we have,

$$|\boldsymbol{U}_m| \leq \sum_{j=1}^{m-1} \alpha_{m,j} |\mathrm{FE}(\boldsymbol{U}_j, k_{m,j})| \overset{k_{m,j} \leq k_*}{\leq} \sum_{j \in [m-1]} \alpha_{m,j} |\boldsymbol{U}_j| \leq \max_{j \in [m-1]} |\boldsymbol{U}_j|.$$

By induction over m, we conclude that $|\boldsymbol{U}_m| \leq |\boldsymbol{U}_1|$, i.e. $\boldsymbol{U}_{s+1} = \boldsymbol{u}_{n+1}$ satisfies $|\boldsymbol{u}_{n+1}| \leq |\boldsymbol{u}_n|$.

c. For the scheme in question, we have,

$$\begin{aligned}
\boldsymbol{U}_1 &= \boldsymbol{u}_n \\
\boldsymbol{U}_2 &= \boldsymbol{u}_n + k\boldsymbol{f}(\boldsymbol{U}_1) \\
&= \boldsymbol{U}_1 + k\boldsymbol{f}(\boldsymbol{U}_1) \\
\boldsymbol{U}_3 &= \boldsymbol{u}_n + \frac{k}{4}\boldsymbol{f}(\boldsymbol{U}_1) + \frac{k}{4}\boldsymbol{f}(\boldsymbol{U}_2) \\
&= \alpha_{3,1}\boldsymbol{U}_1 + (1-\alpha_{3,1})\left(\boldsymbol{U}_2 - k\boldsymbol{f}(\boldsymbol{U}_1)\right) + \frac{k}{4}\boldsymbol{f}(\boldsymbol{U}_1) + \frac{k}{4}\boldsymbol{f}(\boldsymbol{U}_2) \\
&= \alpha_{3,1}\boldsymbol{U}_1 + k\left(-\frac{3}{4} + \alpha_{3,1}\right)\boldsymbol{f}(\boldsymbol{U}_1) + (1-\alpha_{3,1})\boldsymbol{U}_2 + \frac{k}{4}\boldsymbol{f}(\boldsymbol{U}_2).
\end{aligned}$$

where we have introduced a general $0 \leq \alpha_{3,1} \leq 1$ and used $\alpha_{3,2} = 1 - \alpha_{3,1}$. In order for this to satisfy $\beta_{3,1} \geq 0$, then we require,

$$\alpha_{3,1} \geq \frac{3}{4}.$$

The final stage has the form,

$$\begin{aligned}
\boldsymbol{u}_{n+1} &= \boldsymbol{u}_n + \frac{k}{6}\boldsymbol{f}(\boldsymbol{U}_1) + \frac{k}{6}\boldsymbol{f}(\boldsymbol{U}_2) + \frac{2k}{3}\boldsymbol{f}(\boldsymbol{U}_3) \\
&= \alpha_{4,1}\boldsymbol{U}_1 + \alpha_{4,2}(\boldsymbol{U}_2 - k\boldsymbol{f}(U_1)) + (1-\alpha_{4,1}-\alpha_{4,2})\left(\boldsymbol{U}_3 - \frac{k}{4}\boldsymbol{f}(\boldsymbol{U}_1) - \frac{k}{4}\boldsymbol{f}(\boldsymbol{U}_2)\right) \\
&\quad + \frac{k}{6}\boldsymbol{f}(\boldsymbol{U}_1) + \frac{k}{6}\boldsymbol{f}(\boldsymbol{U}_2) + \frac{2k}{3}\boldsymbol{f}(\boldsymbol{U}_3) \\
&= \alpha_{4,1}\boldsymbol{U}_1 + \beta_{4,1}\boldsymbol{f}(\boldsymbol{U}_1) + \alpha_{4,2}\boldsymbol{U}_2 + \beta_{4,2}\boldsymbol{f}(\boldsymbol{U}_2) + (1-\alpha_{4,1}-\alpha_{4,2})\boldsymbol{U}_3 + \beta_{4,3}\boldsymbol{f}(\boldsymbol{U}_3),
\end{aligned}$$

where

$$\begin{aligned}
\beta_{4,1}/k &= -\alpha_{4,2} + \frac{\alpha_{4,1}+\alpha_{4,2}-1}{4} + \frac{1}{6} = -\frac{1}{12} + \frac{1}{4}\alpha_{4,1} - \frac{3}{4}\alpha_{4,2} \\
\beta_{4,2}/k &= \frac{\alpha_{4,1}+\alpha_{4,2}-1}{4} + \frac{1}{6} = -\frac{1}{12} + \frac{1}{4}\alpha_{4,1} + \frac{1}{4}\alpha_{4,2} \\
\beta_{4,3}/k &= \frac{2}{3}.
\end{aligned}$$

In order for these β coefficients to be non-negative, we require that the α coefficients satisfy,

$$\begin{aligned}
\alpha_{4,1} - 3\alpha_{4,2} &\geq \frac{1}{3} \\
\alpha_{4,1} + \alpha_{4,2} &\geq \frac{1}{3}
\end{aligned}$$

This pair of inequalities is satisfied if,

$$\alpha_{4,1} \geq \frac{1}{3},$$
$$\alpha_{4,2} \leq \frac{\alpha_{4,1}}{3} - \frac{1}{9}.$$

Hence, choosing any $\alpha_{3,1} \geq \frac{3}{4}$, $\alpha_{4,1} \geq \frac{1}{3}$, and $\alpha_{4,2} \leq \frac{\alpha_{4,1}}{3} - \frac{1}{9}$ shows that this an SSP scheme (with nonzero c if $\alpha_{4,2} > 0$).

d. In order for the Ralston scheme to be SSP, we see if it can be rewritten as a convex combination of Forward Euler steps:

$$\begin{aligned}
\boldsymbol{U}_1 &= \boldsymbol{u}_n \\
\boldsymbol{U}_2 &= \boldsymbol{u}_n + \frac{2k}{3}\boldsymbol{f}(\boldsymbol{U}_1) = \boldsymbol{U}_1 + \frac{2k}{3}\boldsymbol{f}(\boldsymbol{U}_1) \\
\boldsymbol{u}_{n+1} &= \boldsymbol{u}_n + \frac{k}{4}\boldsymbol{f}(\boldsymbol{U}_1) + \frac{3k}{4}\boldsymbol{f}(\boldsymbol{U}_2) \\
&= \alpha_{3,1}\boldsymbol{U}_1 + (1-\alpha_{3,1})\left(\boldsymbol{U}_2 - \frac{2k}{3}\boldsymbol{f}(\boldsymbol{U}_1)\right) + \frac{k}{4}\boldsymbol{f}(\boldsymbol{U}_1) + \frac{3k}{4}\boldsymbol{f}(\boldsymbol{U}_2) \\
&= \alpha_{3,1}\boldsymbol{U}_1 + k\left[\frac{1}{4} - \frac{2}{3}(1-\alpha_{3,1})\right]\boldsymbol{f}(\boldsymbol{U}_1) + (1-\alpha_{3,1})\boldsymbol{U}_2 + k\frac{3}{4}\boldsymbol{f}(\boldsymbol{U}_2)
\end{aligned}$$

To make this SSP, we require,

$$\frac{1}{4} - \frac{2}{3}(1-\alpha_{3,1}) \geq 0,$$

i.e., $\alpha_{3,1} \geq 5/8$. Hence, this *is* an SSP scheme. The SSP coefficient is the minimum over all the expressions,

$$\begin{aligned}
g_1(\alpha_{3,1}) &= \frac{\alpha_{2,1}}{\beta_{2,1}/k} = \frac{1}{2/3} = \frac{3}{2} \\
g_2(\alpha_{3,1}) &= \frac{\alpha_{3,1}}{\beta_{3,1}/k} = \frac{\alpha_{3,1}}{\frac{1}{4} - \frac{2}{3}(1-\alpha_{3,1})} = \frac{12}{8 - \frac{5}{\alpha_{3,1}}} \\
g_3(\alpha_{3,1}) &= \frac{\alpha_{3,2}}{\beta_{3,2}/k} = \frac{1-\alpha_{3,1}}{3/4} = \frac{4}{3}(1-\alpha_{3,1})
\end{aligned}$$

We seek $\max_{\alpha_{3,1}} \min_{j\in[3]} g_j(\alpha_{3,1})$ as the best (largest) SSP coefficient. First we note that

$$g_2(\alpha_{3,1}) \geq 4 \geq \frac{3}{2} = g_1(\alpha_{3,1}), \qquad \alpha_{3,1} \in [5/8, 1],$$

and hence we may ignore g_2. We also note that,

$$g_1(\alpha_{3,1}) = \frac{3}{2} > \frac{4}{3} \geq \frac{4}{3}(1-\alpha_{3,1}) = g_3(\alpha_{3,1}), \qquad \alpha_{3,1} \in [5/8, 1],$$

and hence we may also ignore g_1. Therefore, we need only maximize the minimum of g_3, which is achieved by:

$$\max_{\alpha_{3,1}\in[5/8,1]} g_2(\alpha_{3,1}) = \frac{1}{2},$$

so the SSP coefficient for this method is $c = 1/2$, which can be realized by taking $\alpha_{3,1} = \frac{5}{8}$.

4. (Exponential Integrators)
For this problem, consider the ODE,

$$\boldsymbol{u}'(t) = \boldsymbol{A}\boldsymbol{u} + \boldsymbol{N}(t, \boldsymbol{u}),$$

where $\boldsymbol{A}$ is a fixed matrix and $\boldsymbol{N}$ is an arbitrary, e.g., nonlinear, function.

a. With initial data $\boldsymbol{u}(0) = \boldsymbol{u}_0$, show that the solution to this IVP at time $t > 0$ is given by,

$$\boldsymbol{u}(t) = e^{t\boldsymbol{A}}\boldsymbol{u}_0 + \int_0^t e^{(t-s)\boldsymbol{A}}\boldsymbol{N}(s, \boldsymbol{u}(s))\,\mathrm{d}s, \tag{3}$$

where $e^{t\boldsymbol{A}}$ is the matrix exponential of $t\boldsymbol{A}$.

b. *Exponential Integrators* form a scheme by setting $(0, t) \leftarrow (t_n, t_{n+1})$, replacing $e^{t\boldsymbol{A}}$ with $e^{(t_{n+1}-t_n)\boldsymbol{A}}$, and discretizing the integral above by approximating $\boldsymbol{N}(\boldsymbol{u}(s))$ with a quadrature rule/polynomial approximation. The matrix exponential term is treated (integrated) exactly. For example, Forward Euler makes the approximation $\boldsymbol{N}(\boldsymbol{u}(s)) \approx \boldsymbol{N}(\boldsymbol{u}_n)$. In terms of matrix operations (possibly including the matrix exponential) write out the Forward Euler and explicit midpoint (RK2) exponential integrator schemes. The explicit midpoint ("modified Euler") scheme has the tableau,

$$\begin{array}{c|cc} 0 & 0 & 0 \\ \frac{1}{2} & \frac{1}{2} & 0 \\ \hline & 0 & 1 \end{array}$$

c. Use both exponential integrator schemes to numerically solve,

$$u_t = u_{xx} + 100u^6(1 - u^6), \qquad\qquad u(x, 0) = \sin \pi x + \frac{1}{4}\sin 2\pi x, \tag{4}$$

with a finite-difference scheme in space for $x \in [0, 1]$ with boundary conditions $u(0) = u(1) = 0$ up to terminal time $T = 1$. Numerically investigate the k-order of convergence of these schemes.

<u>**Solution**</u>:

a. Using that $e^{\boldsymbol{0}} = \boldsymbol{I}$, then the prescribed solution clearly satisfies the initial conditions. To show that it satisfies the ODE, we will recall two facts. The first is the (generalized) Fundamental Theorem of Calculus:

$$\frac{\mathrm{d}}{\mathrm{d}t}\int_{a(t)}^{b(t)} g(t, s)\,\mathrm{d}s = g(t, b(t)) - g(t, a(t)) + \int_{a(t)}^{b(t)} \frac{\partial g}{\partial t}(t, s)\,\mathrm{d}s$$

The second is the defining property of the matrix exponential:

$$\boldsymbol{y}(t) = e^{t\boldsymbol{A}}\boldsymbol{y}_0 \implies \boldsymbol{y}' = \boldsymbol{A}\boldsymbol{y} \implies \frac{\mathrm{d}}{\mathrm{d}t}e^{t\boldsymbol{A}}\boldsymbol{y}_0 = \boldsymbol{A}e^{t\boldsymbol{A}}\boldsymbol{y}_0.$$

Therefore,

$$\begin{aligned} \frac{\mathrm{d}}{\mathrm{d}t}\boldsymbol{u}(t) &= \boldsymbol{A}e^{t\boldsymbol{A}}\boldsymbol{u}_0 + \int_0^t \boldsymbol{A}e^{(t-s)\boldsymbol{A}}\boldsymbol{N}(s, \boldsymbol{u}(s))\,\mathrm{d}s + \boldsymbol{N}(t, \boldsymbol{u}(t)) \\ &= \boldsymbol{A}\left(e^{t\boldsymbol{A}}\boldsymbol{u}_0 + \int_0^t e^{(t-s)\boldsymbol{A}}\boldsymbol{N}(s, \boldsymbol{u}(s))\,\mathrm{d}s\right)\boldsymbol{u}_0 + \boldsymbol{N}(t, \boldsymbol{u}(t)) \\ &= \boldsymbol{A}\boldsymbol{u} + \boldsymbol{N}(t, \boldsymbol{u}(t)), \end{aligned}$$

verifying that $\boldsymbol{u}$ satisfies the ODE.

b. Before tackling this problem, we make another slight digression to establish a useful fact: if, as before $\boldsymbol{y}' = e^{t\boldsymbol{A}}\boldsymbol{y}_0$, then

$$\frac{\mathrm{d}}{\mathrm{d}t} e^{t\boldsymbol{A}}\boldsymbol{y}_0 = \boldsymbol{A}e^{t\boldsymbol{A}}\boldsymbol{y}_0 \quad \Longrightarrow \quad \frac{\mathrm{d}}{\mathrm{d}t}\left(\boldsymbol{A}^{-1}e^{t\boldsymbol{A}}\boldsymbol{y}_0\right) = e^{t\boldsymbol{A}}\boldsymbol{y}_0,$$

establishing that $\boldsymbol{A}^{-1}e^{t\boldsymbol{A}}$ is an antiderivative of $e^{t\boldsymbol{A}}$. Armed with this, the forward Euler discretization of (3) is,

$$\begin{aligned}
\boldsymbol{u}_{n+1} &= e^{k\boldsymbol{A}}\boldsymbol{u}_n + \int_{t_n}^{t_{n+1}} e^{(t_{n+1}-s)\boldsymbol{A}}\boldsymbol{N}(t_n, \boldsymbol{u}_n)\,\mathrm{d}s \\
&= e^{k\boldsymbol{A}}\boldsymbol{u}_n - \boldsymbol{A}^{-1}e^{(t_{n+1}-s)\boldsymbol{A}}\boldsymbol{N}(t_n, \boldsymbol{u}_n)\Big|_{s=t_n}^{t_n+k} \\
&= e^{k\boldsymbol{A}}\boldsymbol{u}_n - \boldsymbol{A}^{-1}\boldsymbol{N}(t_n, \boldsymbol{u}_n) - \boldsymbol{A}^{-1}e^{k\boldsymbol{A}}\boldsymbol{N}(t_n, \boldsymbol{u}_n) \\
&= e^{k\boldsymbol{A}}\boldsymbol{u}_n - \boldsymbol{A}^{-1}\left(\boldsymbol{I} - e^{k\boldsymbol{A}}\right)\boldsymbol{N}(t_n, \boldsymbol{u}_n).
\end{aligned}$$

where we have used the standard notation $t_n = nk$. The modified Euler (RK2) scheme is given by a Forward Euler intermediate step of size $k/2$, followed by a full Forward Euler step using the intermediate stage as the approximation to $\boldsymbol{u}$:

$$\begin{aligned}
\boldsymbol{U}_1 &= e^{0\boldsymbol{A}}\boldsymbol{u}_n + \int_{t_n}^{t_n} e^{(t-s)\boldsymbol{A}}\boldsymbol{N}(s, \boldsymbol{u}(s))\,\mathrm{d}s = \boldsymbol{u}_n \\
\boldsymbol{U}_2 &= e^{\frac{k}{2}\boldsymbol{A}}\boldsymbol{u}_n + \int_{t_n}^{t_n+\frac{k}{2}} e^{(t-s)\boldsymbol{A}}\boldsymbol{N}(t_n, \boldsymbol{U}_1)\,\mathrm{d}s = e^{\frac{k}{2}\boldsymbol{A}}\boldsymbol{u}_n - \boldsymbol{A}^{-1}\left(\boldsymbol{I} - e^{\frac{k}{2}\boldsymbol{A}}\right)\boldsymbol{N}(t_n, \boldsymbol{u}_n) \\
\boldsymbol{u}_{n+1} &= e^{k\boldsymbol{A}}\boldsymbol{u}_n + \int_{t_n}^{t_n+k} e^{(t-s)\boldsymbol{A}}\boldsymbol{N}(t_{n+1/2}, \boldsymbol{U}_2)\,\mathrm{d}s = e^{k\boldsymbol{A}}\boldsymbol{u}_n - \boldsymbol{A}^{-1}\left(\boldsymbol{I} - e^{k\boldsymbol{A}}\right)\boldsymbol{N}(t_n + k/2, \boldsymbol{U}_2)
\end{aligned}$$

c. We implement both of these schemes using the spatial discretization,

$$\frac{\mathrm{d}}{\mathrm{d}t}u_j(t) = D_+D_-u_j(t) + 100u_j(t)^6(1 - u_j(t)^6), \qquad\qquad j \in [M],$$

on an equispaced grid of $M = 100$ interior points over $[0, 1]$. We report errors using both exponential integrator schemes using $k = T/N$ for increasing choices of N. We show results in table 1, where orders/rates of convergence between simulations with $k = k_1$ and $k = k_2$ are computed as,

$$\text{rate} = \frac{\log\left(\frac{\text{err}(k_1)}{\text{err}(k_2)}\right)}{\log\left(\frac{k_1}{k_2}\right)}.$$

Errors are computed as $\sqrt{h}$-scaled vector ℓ^2 errors at the terminal time:

$$\text{error} = \sqrt{h}\sqrt{\sum_{j=1}^{M}\left(u_j^N - U_j(T)\right)^2},$$

where $U_j(T)$ is the solution computed using an explicit fourth-order Runge-Kutta method with step size $k = 2 \times 10^{-5} = \frac{1}{50000}$. We see from table 1 that the exponential Euler method outperforms its RK2 variant for relatively small values of k, but they both reach

$k = \frac{1}{N}$	Exp Euler error	Exp Euler rate	Exp RK2 error	Exp RK2 rate
$\frac{1}{200}$	4.46×10^{-2}	—	1.19×10^{-1}	—
$\frac{1}{250}$	1.28×10^{-2}	5.58	1.35×10^{-1}	-0.58
$\frac{1}{300}$	1.66×10^{-13}	137.50	1.48×10^{-1}	-0.50
$\frac{1}{350}$	1.67×10^{-13}	-0.06	1.59×10^{-1}	-0.43
$\frac{1}{400}$	1.69×10^{-13}	-0.03	1.67×10^{-1}	-0.37
$\frac{1}{450}$	1.70×10^{-13}	-0.05	1.73×10^{-1}	-0.31
$\frac{1}{500}$	1.71×10^{-13}	-0.09	1.71×10^{-13}	262.34

Table 1: Errors and orders of convergence for Exponential Euler and Exponential RK2 methods for problem (4).

machine precision error for sufficiently large k. Note that this type of behavior is not surprising since this is a (very) nonlinear problem; the expected orders of convergence for both of these methods is in the k-asymptotic regime only, but anything can happen in the pre-asymptotic regime. The main advantage of using the exponential integrators is that they handle the stiff term u_{xx} very well. Using the standard explicit Runge-Kutta 4 requires $N \gtrsim 14500$ before the numerical solution is even stable. Hence, exponential integrators can reduce the timestep restriction in this case by almost an order of magnitude.

5. (Well-posed linear PDEs)
Consider the IVP,

$$u_t = 3u_x - u_{xx} - u_{xxxx}, \qquad u(x,0) = u_0(x), \tag{5}$$

with periodic boundary conditions on $x \in [0, 2\pi)$.

a. Determine if the PDE is well-posed in the sense of the definition on slide `D10-S05`.

b. Compute the exact solution to this PDE.

Solution:

a. The symbol of this PDE is,

$$\begin{aligned}\mathcal{P}(\omega) &= \mathcal{F}\left\{3\frac{\partial}{\partial x} - \frac{\partial^2}{\partial x^2} - \frac{\partial^4}{\partial x^4}\right\} \\ &= 3i\omega + \omega^2 - \omega^4.\end{aligned}$$

To assess well-posedness, we evaluate,

$$\left|e^{\mathcal{P}(\omega)}\right| = e^{\omega^2 - \omega^4} \leq 1,$$

for any $\omega \in \mathbb{Z}$. Hence, this PDE is well-posed.

b. Since the PDE is well-posed, we can compute the exact solution through a Fourier Series approach. Taking the Fourier transform of the PDE yields:

$$\frac{\mathrm{d}}{\mathrm{d}t}U(\omega,t) = \mathcal{P}(\omega)U(\omega,t), \qquad \omega \in \mathbb{Z},$$

with initial conditions $U(\omega, 0)$ defined by,

$$u_0(x) = \sum_{\omega \in \mathbb{Z}} U_0(\omega) e^{i\omega x}.$$

The solution to the ODE system is therefore,

$$U(\omega, t) = e^{P(\omega)t} U_0(\omega) = e^{3i\omega t + (\omega^2 - \omega^4)t} U_0(\omega).$$

Hence, the full solution is,

$$u(x, t) = \sum_{\omega \in \mathbb{Z}} U(\omega, t) e^{i\omega x} = \sum_{\omega \in \mathbb{Z}} e^{3i\omega t + (\omega^2 - \omega^4)t} e^{i\omega x} U_0(\omega)$$