Runge-Kutta/‘Single-step’/‘Multi-Stage’ Methods

**Idea:** Compute intermediate ‘stage values’:

\[
\begin{align*}
    r_1 &= f(t_k + c_1 h, y_k + (a_{11} \cdot r_1 + \cdots + a_{1s} \cdot r_s) h) \\
    &\vdots \\
    r_s &= f(t_k + c_s h, y_k + (a_{s1} \cdot r_1 + \cdots + a_{ss} \cdot r_s) h)
\end{align*}
\]

Then compute the new state from those:

\[
y_{k+1} = y_k + (b_1 \cdot r_1 + \cdots + b_s \cdot r_s) h
\]

Can summarize in a **Butcher tableau**:

\[
\begin{array}{c|cccc}
    c_1 & a_{11} & \cdots & a_{1s} \\
    \vdots & \vdots & \ddots & \vdots \\
    c_s & a_{s1} & \cdots & a_{ss} \\
    \hline
    b_1 & \cdots & b_s
\end{array}
\]

- When is an RK method explicit?

If only \(a_{ij}\) below the diagonal are non-zero.
o When is it implicit?

o When is it **diagonally implicit**? (And what does that mean?)

o Stuff Heun’s method into a Butcher tableau:

1. $\tilde{y}_{k+1} = y_k + hf(y_k)$

2. $y_{k+1} = y_k + \frac{h}{2}(f(y_k) + f(\tilde{y}_{k+1}))$.  

o What is RK4?

\[

t_i = f(y_n)  \\
\tilde{r}_2 = f(\tilde{y}_{n+1}) = f(y_n + hf(y_n))  \\
= f(y_n + h r_1)
\]
**Multi-step/Single-stage/Adams Methods/Backward Differencing Formulas (BDFs)**

**Idea:** Instead of computing stage values, use *history* (of either values of \( f \) or \( y \) or both):

\[
y_{k+1} = \sum_{i=1}^{M} \alpha_i y_{k+1-i} + h \sum_{i=1}^{N} \beta_i f(y_{k+1-i})
\]

(one of these \( \rightarrow \) hw) Extensions to implicit possible.

- Method relies on existence of history. What if there isn’t any? (Such as at the start of time integration?)
Demo: Stability regions
10 Boundary Value Problems for ODEs
BVP Problem Setup: Second Order

Example: Second-order linear ODE

\[ u''(x) + p(x)u'(x) + q(x)u(x) = r(x) \]

with boundary conditions (‘BCs’) at \( a \):

- **Dirichlet** \( u(a) = u_a \)
- or **Neumann** \( u'(a) = v_a \)
- or **Robin** \( \alpha u(a) + \beta u'(a) = w_a \)

and the same choices for the BC at \( b \).

*Note: BVPs in time are rare in applications, hence \( x \) (not \( t \)) is typically used for the independent variable.*
BVP Problem Setup: General Case

ODE:
\[ y'(x) = f(y(x)) \quad f: \mathbb{R}^n \rightarrow \mathbb{R}^n \]

BCs:
\[ g(y(a), y(b)) = 0 \quad g: \mathbb{R}^{2n} \rightarrow \mathbb{R}^n \]

(Recall the rewriting procedure to first-order for any-order ODEs.)

- Does a first-order, scalar BVP make sense?

**Example:** Linear BCs
\[ B_a y(a) + B_b y(b) = c \]

- Is this Dirichlet/Neumann/...?
10.1 Existence, Uniqueness, Conditioning
Does a solution even exist? How sensitive are they?

General case is harder than root finding, and we couldn’t say much there. → Only consider linear BVP.

\[
\begin{align*}
(*) \quad & \begin{cases}
y'(x) = A(x)y(x) + b(x) \\
B_ay(a) + B_by(b) = c
\end{cases}
\end{align*}
\]

To solve that, consider homogeneous IVP

\[
y_i'(x) = A(x)y_i(x)
\]

with initial condition

\[
y_i(a) = e_i.
\]

Note: \( y \neq y_i \). \( e_i \) is the \( i \)th unit vector. With that, build fundamental solution matrix

\[
Y(x) = \begin{pmatrix}
y_1 \\
\vdots \\
y_n
\end{pmatrix}
\]
Let

\[ Q := B_a Y(a) + B_b Y(b) \]

Then (*) has a unique solution if and only if \( Q \) is invertible. Solve to find coefficients:

\[ Q\alpha = c \]

Then \( Y(x)\alpha \) solves (*) with \( b(x) = 0 \).

Define \( \Phi(x) := Y(x)Q^{-1} \). So \( \Phi(x)c \) solves (*) with \( b(x) = 0 \).
Define Green’s function

\[ G(x, y) := \begin{cases} \Phi(x) B_a \Phi(a) \Phi^{-1}(y) & y \leq x, \\ -\Phi(x) B_b \Phi(b) \Phi^{-1}(y) & y > x. \end{cases} \]

Then

\[ y(x) = \Phi(x) c + \int_a^b G(x, y) b(y) dy. \]

**Conditioning:**

Now easy. For perturbed problem with \( b(x) + \Delta b(x) \) and \( c + \Delta c \):

\[ \| \Delta y \|_\infty \leq \max (\| \Phi \|_\infty, \| G \|_\infty) \left( \| \Delta c \|_1 + \int \| \Delta b(y) \|_1 dy \right). \]

- Did not prove uniqueness. (But true.)
- Also get continuous dependence on data.
- Can verify that above formula solves (*) by plug’n’chug.
10.2 Numerical Methods
Shooting Method

**Idea:** Want to make use of the fact that we can already solve IVPs.

**Problem:** Don’t know *all* left BCs.

**Demo:** Shooting Method

- What about systems?

- What are some downsides of this method?

- What’s an alternative approach?