Single-Step Methods and Runge-Kutta Methods

Single-Step Methods

Single-step methods are used to solve first-order ordinary differential equations of the form:

$$\frac{dy}{dx} = f(x, y), \quad y(x_0) = y_0 \tag{1}$$

The general form of a single-step method is:

$$y_{n+1} = y_n + h\phi(x_n, y_n, h) \tag{2}$$

where h is the step size and ϕ is the increment function that determines the method's characteristics.

Properties

- Self-starting: Only requires the current point (x_n, y_n) to compute y_{n+1}
- Easy to implement variable step sizes
- Examples: Euler's method, Runge-Kutta methods, Taylor series methods

Runge-Kutta Methods: General Framework

General s-stage Runge-Kutta Method

An s-stage Runge-Kutta method has the form:

$$y_{n+1} = y_n + h \sum_{i=1}^{s} w_i k_i \tag{3}$$

where

$$k_i = f\left(x_n + c_i h, y_n + h \sum_{j=1}^{s} a_{ij} k_j\right), \quad i = 1, 2, \ldots, s \tag{4}$$

The Butcher Tableau

The method is completely characterized by the coefficients arranged in the **Butcher tableau**:

$$
\begin{array}{c|c} \mathbf{c} & \mathbf{A} \\ \hline & \mathbf{w}^T \end{array} \quad = \quad \begin{array}{c|cccc} c_1 & a_{11} & a_{12} & \cdots & a_{1s} \\ c_2 & a_{21} & a_{22} & \cdots & a_{2s} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ c_s & a_{s1} & a_{s2} & \cdots & a_{ss} \\ \hline & w_1 & w_2 & \cdots & w_s \end{array} \tag{5}
$$

where:

- $\mathbf{A} = [a_{ij}]$ is the **coefficient matrix** (or Runge-Kutta matrix)
- $\mathbf{w} = [w_1, w_2, \ldots, w_s]^T$ is the **weight vector**
- $\mathbf{c} = [c_1, c_2, \ldots, c_s]^T$ is the **node vector**

The **consistency conditions** for a Runge-Kutta method are:

$$
\sum_{i=1}^{s} w_i = 1 \text{ and } c_i = \sum_{j=1}^{s} a_{ij}, \quad i = 1, 2, \ldots, s
$$

In compact form:

$$
w^T e = 1, \quad \text{and} \quad c = Ae, \text{ where } e = [1, 1, \ldots, 1]^T \in \mathbb{R}^s \text{ is the vector of ones.}
$$

These conditions ensure that the RK method is **at least first-order accurate**.

The Coefficient Matrix A

The coefficient matrix **A** determines the structure and properties of the method:

- Explicit Methods If **A** is **strictly lower triangular** (all $a_{ij} = 0$ for $j \geq i$):

$$
k_i = f\left(x_n + c_i h, y_n + h \sum_{j=1}^{i-1} a_{ij} k_j\right) \tag{6}
$$

  Each k_i can be computed directly from previously computed stages. No system of equations needs to be solved.

- Implicit Methods If **A** has non-zero diagonal or upper triangular elements, the stages k_i are coupled and require solving a system of (possibly nonlinear) equations simultaneously.

Second-Order Runge-Kutta (RK2)

- Modified Euler Cauchy

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

$$y_{n+1} = y_n + k_2 \tag{7}$$
$$k_1 = hf(x_n, y_n) \tag{8}$$
$$k_2 = hf\left(x_n + \frac{h}{2}, y_n + \frac{k_1}{2}\right) \tag{9}$$

- Euler-Cauchy or Heun's Method

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

$$k_1 = hf(x_n, y_n) \tag{10}$$
$$k_2 = hf(x_n + h, y_n + k_1) \tag{11}$$
$$y_{n+1} = y_n + \frac{k_1 + k_2}{2} \tag{12}$$

- Optimal

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

Properties

- Explicit method (**A** is strictly lower triangular)
- Two function evaluations per step

Third-Order Runge-Kutta (RK3)

- Classical

$$\begin{array}{c|ccc} 0 & 0 & 0 & 0 \\ \frac{1}{2} & \frac{1}{2} & 0 & 0 \\ 1 & -1 & 2 & 0 \\ \hline & \frac{1}{6} & \frac{2}{3} & \frac{1}{6} \end{array} \tag{13}$$

$$k_1 = hf(x_n, y_n) \tag{14}$$

$$k_2 = hf\left(x_n + \frac{h}{2}, y_n + \frac{k_1}{2}\right) \tag{15}$$

$$k_3 = hf(x_n + h, y_n - k_1 + 2k_2) \tag{16}$$

$$y_{n+1} = y_n + \frac{1}{6}(k_1 + 4k_2 + k_3) \tag{17}$$

- Heun

$$\begin{array}{c|ccc} 0 & 0 & 0 & 0 \\ \frac{1}{3} & \frac{1}{3} & 0 & 0 \\ \frac{2}{3} & 0 & \frac{2}{3} & 0 \\ \hline & \frac{1}{4} & 0 & \frac{3}{4} \end{array} \tag{18}$$

- Nystrom

$$\begin{array}{c|ccc} 0 & 0 & 0 & 0 \\ 2/3 & 2/3 & 0 & 0 \\ 2/3 & 0 & 2/3 & 0 \\ \hline & 2/8 & 3/8 & 3/8 \end{array} \tag{19}$$

- Nearly optimal

$$\begin{array}{c|ccc} 0 & 0 & 0 & 0 \\ 1/2 & 1/2 & 0 & 0 \\ 3/4 & 0 & 3/4 & 0 \\ \hline & 2/9 & 3/9 & 4/9 \end{array} \tag{20}$$

  Properties

  - Explicit method
  - Three function evaluations per step

Fourth-Order Runge-Kutta (RK4)

- Classical

$$\begin{array}{c|cccc} 0 & 0 & 0 & 0 & 0 \\ \frac{1}{2} & \frac{1}{2} & 0 & 0 & 0 \\ \frac{1}{2} & 0 & \frac{1}{2} & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 \\ \hline & \frac{1}{6} & \frac{1}{3} & \frac{1}{3} & \frac{1}{6} \end{array} \tag{21}$$

$$k_1 = hf(x_n, y_n) \tag{22}$$

$$k_2 = hf\left(x_n + \frac{h}{2}, y_n + \frac{k_1}{2}\right) \tag{23}$$

$$k_3 = hf\left(x_n + \frac{h}{2}, y_n + \frac{k_2}{2}\right) \tag{24}$$

$$k_4 = hf(x_n + h, y_n + k_3) \tag{25}$$

$$y_{n+1} = y_n + \frac{1}{6}(k_1 + 2k_2 + 2k_3 + k_4) \tag{26}$$

- Kutta

$$\begin{array}{c|cccc} 0 & 0 & 0 & 0 & 0 \\ 1/3 & 1/3 & 0 & 0 & 0 \\ 2/3 & -13 & 1 & 0 & 0 \\ 1 & 1 & -1 & 1 & 0 \\ \hline & 1/8 & 3/8 & 3/8 & 1/8 \end{array} \tag{27}$$

- Properties
  - Explicit method ($\mathbf{A}$ is strictly lower triangular with zeros on diagonal)
  - Four function evaluations per step

Comparison Table

Method	Stages s	Function Evals	Global Error	Stability
RK2	2	2	$O(h^2)$	Conditionally stable
RK3	3	3	$O(h^3)$	Conditionally stable
RK4	4	4	$O(h^4)$	Conditionally stable

Stability Regions

For the test equation $y' = \lambda y$ with $\lambda < 0$, the stability condition is:

$$\left|1 + \lambda h + \frac{(\lambda h)^2}{2!} + \cdots\right| \leq 1 \tag{28}$$

- RK2: stability region slightly larger than Euler
- RK4: larger stability region, allows larger step sizes
- All explicit RK methods have bounded stability regions

Key Points

1. **Coefficient Matrix A** determines whether the method is explicit, implicit, or semi-explicit

2. **RK4** is the industry standard due to excellent balance of accuracy and computational cost

3. All classical RK methods (RK2, RK3, RK4) are explicit with **A** strictly lower triangular