Runge-Kutta methods

With orders of Taylor methods yet without derivatives of $f(t, y(t))$

First order Taylor expansion in two variables

Theorem: Suppose that $f(t, y)$ and all its partial derivatives are

continuous on $\quad D \overset{def}{=} \{(t, y) \,|\, \quad a \leq t \leq b, \quad c \leq y \leq d.\}$

Let $(t_0, y_0), (t, y) \in D, \Delta_t = t - t_0, \Delta_y = y - y_0$. **Then**

$$f(t, y) = P_1(t, y) + R_1(t, y), \quad \text{where}$$

First order Taylor expansion in two variables

Theorem: Suppose that $f(t,y)$ and all its partial derivatives are

continuous on $D \stackrel{def}{=} \{(t,y) \,|\, \quad a \leq t \leq b, \quad c \leq y \leq d.\}$

Let $(t_0, y_0), (t, y) \in D, \Delta_t = t - t_0, \Delta_y = y - y_0$. **Then**

$$f(t,y) = P_1(t,y) + R_1(t,y), \quad \text{where}$$

$$\begin{aligned} P_1(t,y) &= f(t_0,y_0) + \Delta_t \frac{\partial f}{\partial t}(t_0,y_0) + \Delta_y \frac{\partial f}{\partial y}(t_0,y_0), \\ R_1(t,y) &= \frac{\Delta_t^2}{2}\frac{\partial^2 f}{\partial t^2}(\xi,\mu) + \Delta_t\,\Delta_y \frac{\partial^2 f}{\partial t \partial y}(\xi,\mu) + \frac{\Delta_y^2}{2}\frac{\partial^2 f}{\partial y^2}(\xi,\mu), \end{aligned}$$

for some point $(\xi, \mu) \in D$.

Second order Taylor expansion in two variables

Theorem: Suppose that $f(t, y)$ and all its partial derivatives are

$$\text{continuous on} \quad D \stackrel{def}{=} \{(t, y) \,|\quad a \leq t \leq b, \quad c \leq y \leq d.\}$$

Let $(t_0, y_0), (t, y) \in D, \Delta_t = t - t_0, \Delta_y = y - y_0$. **Then**

$$f(t, y) = P_2(t, y) + R_2(t, y), \quad \text{where}$$

Second order Taylor expansion in two variables

Theorem: Suppose that $f(t, y)$ and all its partial derivatives are

continuous on $\quad D \stackrel{def}{=} \{(t, y) \,|\quad a \leq t \leq b, \quad c \leq y \leq d.\}$

Let $(t_0, y_0), (t, y) \in D, \Delta_t = t - t_0, \Delta_y = y - y_0$. **Then**

$$f(t, y) = P_2(t, y) + R_2(t, y), \quad \text{where}$$

$$\begin{aligned} P_2(t, y) \quad &= \quad f(t_0, y_0) + \left(\Delta_t \frac{\partial f}{\partial t}(t_0, y_0) + \Delta_y \frac{\partial f}{\partial y}(t_0, y_0)\right) \\ &+ \left(\frac{\Delta_t^2}{2} \frac{\partial^2 f}{\partial t^2}(t_0, y_0) + \Delta_t \, \Delta_y \frac{\partial^2 f}{\partial t \partial y}(t_0, y_0) + \frac{\Delta_y^2}{2} \frac{\partial^2 f}{\partial y^2}(t_0, y_0)\right), \end{aligned}$$

Second order Taylor expansion in two variables

Theorem: Suppose that $f(t, y)$ and all its partial derivatives are

continuous on $D \stackrel{def}{=} \{(t, y) \,|\, \quad a \le t \le b, \quad c \le y \le d.\}$

Let $(t_0, y_0), (t, y) \in D, \Delta_t = t - t_0, \Delta_y = y - y_0$. **Then**

$$f(t, y) = P_2(t, y) + R_2(t, y), \quad \text{where}$$

$$\begin{aligned} P_2(t, y) &= f(t_0, y_0) + \left(\Delta_t \frac{\partial f}{\partial t}(t_0, y_0) + \Delta_y \frac{\partial f}{\partial y}(t_0, y_0)\right) \\ &\quad + \left(\frac{\Delta_t^2}{2} \frac{\partial^2 f}{\partial t^2}(t_0, y_0) + \Delta_t \, \Delta_y \frac{\partial^2 f}{\partial t \partial y}(t_0, y_0) + \frac{\Delta_y^2}{2} \frac{\partial^2 f}{\partial y^2}(t_0, y_0)\right), \\ R_2(t, y) &= \frac{1}{3!} \left(\sum_{j=0}^{3} \begin{pmatrix} 3 \\ j \end{pmatrix} \Delta_t^{3-j} \, \Delta_y^{j} \frac{\partial^3 f}{\partial t^{3-j} \partial y^{j}}(\xi, \mu)\right) \end{aligned}$$

Runge-Kutta Method of Order Two (III)

- **Midpoint Method**

$$
\begin{aligned}
w_0 &= \alpha, \\
w_{j+1} &= w_j + h\, f\left(t_j + \frac{h}{2}, w_j + \frac{h}{2} f(t_j, w_j)\right), \quad j = 0, 1, \cdots, N-1.
\end{aligned}
$$

- Two function evaluations for each j,
- Second order accuracy.

No need for derivative calculations

General 2nd order Runge-Kutta Methods

- $w_0 = \alpha$; for $j = 0, 1, \cdots, N-1$,

$$w_{j+1} = w_j + h\left(a_1 f(t_j, w_j) + a_2 f\left(t_j + \alpha_2, w_j + \delta_2 f(t_j, w_j)\right)\right).$$

General 2nd order Runge-Kutta Methods

- $w_0 = \alpha$; for $j = 0, 1, \cdots, N-1$,

$$w_{j+1} = w_j + h\left(a_1 f(t_j, w_j) + a_2 f\left(t_j + \alpha_2, w_j + \delta_2 f(t_j, w_j)\right)\right).$$

- Two function evaluations for each j,
- Want to choose $a_1, a_2, \alpha_2, \delta_2$ for highest possible order of accuracy.

General 2nd order Runge-Kutta Methods

- $w_0 = \alpha$; for $j = 0, 1, \cdots, N-1$,

$$w_{j+1} = w_j + h\left(a_1 f(t_j, w_j) + a_2 f\left(t_j + \alpha_2, w_j + \delta_2 f(t_j, w_j)\right)\right).$$

- Two function evaluations for each j,
- Want to choose $a_1, a_2, \alpha_2, \delta_2$ for highest possible order of accuracy.

local truncation error

$$\begin{aligned}
\tau_{j+1}(h) = \frac{y(t_{j+1}) - y(t_j)}{h} - \left(a_1 f(t_j, y(t_j)) + a_2 f\left(t_j + \alpha_2, y(t_j) + \delta_2 f(t_j, y(t_j))\right)\right) \\
= \; y'(t_j) + \frac{h}{2} y''(t_j) + O(h^2) \\
- \Bigg((a_1 + a_2)\, f(t_j, y(t_j)) + a_2\, \alpha_2\, \frac{\partial f}{\partial t}(t_j, y(t_j)) \\
+ \; a_2\, \delta_2 f(t_j, y(t_j)) \frac{\partial f}{\partial y}(t_j, y(t_j)) + O(h^2) \Bigg).
\end{aligned}$$

local truncation error

$$
\begin{aligned}
\tau_{j+1}(h) = \quad & y'(t_j) + \frac{h}{2} y''(t_j) + O(h^2) \\
& - \Bigg((a_1 + a_2)\, f(t_j, y(t_j)) + a_2\, \alpha_2\, \frac{\partial f}{\partial t}(t_j, y(t_j)) \\
& \qquad + \quad a_2\, \delta_2 f(t_j, y(t_j)) \frac{\partial f}{\partial y}(t_j, y(t_j)) + O(h^2) \Bigg) .
\end{aligned}
$$

where $y'(t_j) = f(t_j, y(t_j))$,

local truncation error

$$\begin{aligned}\tau_{j+1}(h) = & \; y'(t_j) + \frac{h}{2} y''(t_j) + O(h^2) \\ & - \Big((a_1 + a_2)\, f(t_j, y(t_j)) + a_2\, \alpha_2 \frac{\partial f}{\partial t}(t_j, y(t_j)) \\ & \quad + \; a_2\, \delta_2 f(t_j, y(t_j)) \frac{\partial f}{\partial y}(t_j, y(t_j)) + O(h^2) \Big) .\end{aligned}$$

where $y'(t_j) = f(t_j, y(t_j))$,

$$y''(t_j) = \frac{d\,f}{d\,t}(t_j, y(t_j)) = \frac{\partial f}{\partial t}(t_j, y(t_j)) + f(t_j, y(t_j)) \frac{\partial f}{\partial y}(t_j, y(t_j)).$$

For any choice with

$$a_1 + a_2 = 1, \quad a_2\, \alpha_2 = a_2\, \delta_2 = \frac{h}{2},$$

we have a second order method

$$\tau_{j+1}(h) = O(h^2).$$

local truncation error

$$\tau_{j+1}(h) = \quad y'(t_j) + \frac{h}{2} y''(t_j) + O(h^2)$$
$$- \Bigg((a_1 + a_2)\, f(t_j, y(t_j)) + a_2\, \alpha_2\, \frac{\partial f}{\partial t}(t_j, y(t_j))$$
$$+ \quad a_2\, \delta_2 f(t_j, y(t_j)) \frac{\partial f}{\partial y}(t_j, y(t_j)) + O(h^2) \Bigg) .$$

where $y'(t_j) = f(t_j, y(t_j))$,

$$y''(t_j) = \frac{d\, f}{d\, t}(t_j, y(t_j)) = \frac{\partial f}{\partial t}(t_j, y(t_j)) + f(t_j, y(t_j)) \frac{\partial f}{\partial y}(t_j, y(t_j)).$$

For any choice with

$$a_1 + a_2 = 1, \quad a_2\, \alpha_2 = a_2\, \delta_2 = \frac{h}{2},$$

we have a second order method

$$\tau_{j+1}(h) = O(h^2).$$

Four parameters, three equations

General 2nd order Runge-Kutta Methods

$w_0 = \alpha$; for $j = 0, 1, \cdots, N-1$,

$$w_{j+1} = w_j + h\left(a_1 f(t_j, w_j) + a_2 f\left(t_j + \alpha_2, w_j + \delta_2 f(t_j, w_j)\right)\right).$$

$$a_1 + a_2 = 1, \quad a_2\,\alpha_2 = a_2\,\delta_2 = \frac{h}{2},$$

- **Midpoint method**: $a_1 = 0, a_2 = 1, \alpha_2 = \delta_2 = \frac{h}{2}$,

$$w_{j+1} = w_j + h\,f\left(t_j + \frac{h}{2}, w_j + \frac{h}{2} f(t_j, w_j)\right).$$

- **Modified Euler method**: $a_1 = a_2 = \frac{1}{2}, \alpha_2 = \delta_2 = h$,

$$w_{j+1} = w_j + \frac{h}{2}\left(f(t_j, w_j) + f\left(t_{j+1}, w_j + hf(t_j, w_j)\right)\right).$$

3rd order Runge-Kutta Method (rarely used in practice)

$$w_0 = \alpha;$$
for $j = 0, 1, \cdots, N-1,$

$$w_{j+1} = w_j + \frac{h}{4}\left(f(t_j, w_j) + 3f\left(t_j + \frac{2h}{3}, w_j + \frac{2h}{3} f(t_j + \frac{h}{3}, w_j + \frac{h}{3} f(t_j, w_j))\right)\right)$$
$$\stackrel{def}{=} w_j + h\,\phi(t_j, w_j).$$

3rd order Runge-Kutta Method (rarely used in practice)

$$w_0 = \alpha;$$
for $\quad j = 0, 1, \cdots, N-1,$

$$w_{j+1} = w_j + \frac{h}{4}\left(f(t_j, w_j) + 3f\left(t_j + \frac{2h}{3}, w_j + \frac{2h}{3} f(t_j + \frac{h}{3}, w_j + \frac{h}{3} f(t_j, w_j))\right)\right)$$
$$\stackrel{def}{=} w_j + h\,\phi(t_j, w_j).$$

local truncation error

$$\tau_{j+1}(h) = \frac{y(t_{j+1}) y(t_j)}{h} - \phi(t_j, y(t_j)) = O(h^3).$$

4th order Runge-Kutta Method

$w_0 = \alpha;$
for $\quad j = 0, 1, \cdots, N-1,$

$$\begin{aligned}
k_1 &= h\, f(t_j, w_j), \\
k_2 &= h\, f\left(t_j + \frac{h}{2}, w_j + \frac{1}{2} k_1\right), \\
k_3 &= h\, f\left(t_j + \frac{h}{2}, w_j + \frac{1}{2} k_2\right), \\
k_4 &= h\, f\left(t_{j+1}, w_j + k_3\right), \\
w_{j+1} &= w_j + \frac{1}{6}\left(k_1 + 2k_2 + 2k_3 + k_4\right).
\end{aligned}$$

4 function evaluations per step

Example

Initial Value ODE	$\frac{dy}{dt} = y - t^2 + 1, \quad 0 \leq t \leq 2, \quad y(0) = 0.5,$
exact solution	$y(t) = (1 + t)^2 - 0.5\, e^t.$

Example

Initial Value ODE $\quad \frac{dy}{dt} = y - t^2 + 1, \quad 0 \leq t \leq 2, \quad y(0) = 0.5,$

exact solution $\quad y(t) = (1+t)^2 - 0.5\,e^t.$

t_i	Exact	Euler $h = 0.025$	Modified Euler $h = 0.05$	Runge-Kutta Order Four $h = 0.1$
0.0	0.5000000	0.5000000	0.5000000	0.5000000
0.1	0.6574145	0.6554982	0.6573085	0.6574144
0.2	0.8292986	0.8253385	0.8290778	0.8292983
0.3	1.0150706	1.0089334	1.0147254	1.0150701
0.4	1.2140877	1.2056345	1.2136079	1.2140869
0.5	1.4256394	1.4147264	1.4250141	1.4256384

Example

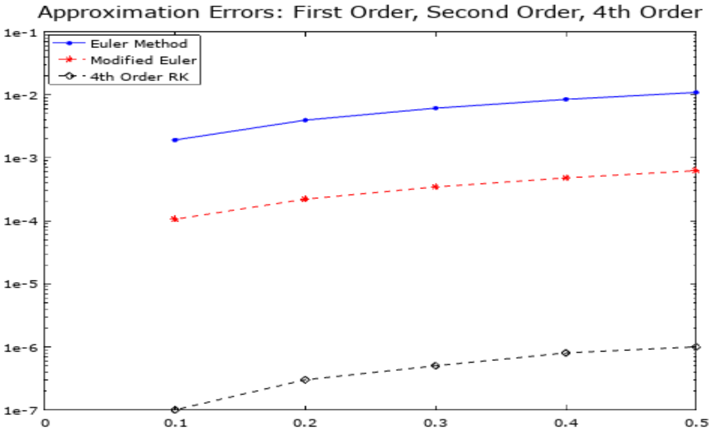
Initial Value ODE $\quad \dfrac{dy}{dt} = y - t^2 + 1, \quad 0 \le t \le 2, \quad y(0) = 0.5,$

exact solution $\quad y(t) = (1+t)^2 - 0.5\, e^t.$

Example

Initial Value ODE $\quad \dfrac{dy}{dt} = y - t^2 + 1, \quad 0 \le t \le 2, \quad y(0) = 0.5,$

exact solution $\quad y(t) = (1+t)^2 - 0.5\, e^t.$



Adaptive Error Control (I)

$$\frac{dy}{dt} = f(t, y(t)), \quad a \leq t \leq b, \quad y(a) = \alpha.$$

Consider a variable-step method with a well-chosen function $\phi(t, w, h)$:

- $w_0 = \alpha$.
- for $j = 0, 1, \cdots$,
  - **choose** step-size $h_j = t_{j+1} - t_j$,
  - **set** $w_{j+1} = w_j + h_j \, \phi\,(t_j, w_j, h_j)$.

Adaptive Error Control (I)

$$\frac{dy}{dt} = f(t, y(t)), \quad a \le t \le b, \quad y(a) = \alpha.$$

Consider a variable-step method with a well-chosen function $\phi(t, w, h)$:

- $w_0 = \alpha$.
- for $j = 0, 1, \cdots,$
  - **choose** step-size $h_j = t_{j+1} - t_j$,
  - **set** $w_{j+1} = w_j + h_j \, \phi\left(t_j, w_j, h_j\right)$.

Adaptively choose step-size to satisfy given tolerance

Adaptive Error Control (II)

- Given an **order-n** method
  - $w_0 = \alpha$.
  - for $j = 0, 1, \cdots$,

$$w_{j+1} = w_j + h\,\phi\left(t_j, w_j, h\right), \quad h = t_{j+1} - t_j,$$

  - **local truncation error** (LTE)

$$\tau_{j+1}(h) = \frac{y(t_{j+1}) - y(t_j)}{h} - \phi\left(t_j, y(t_j), h\right) = O\left(h^n\right).$$

- Given tolerance $\tau > 0$, we would like to estimate **largest** step-size h for which

$$|\tau_{j+1}(h)| \lesssim \tau.$$

Adaptive Error Control (II)

- Given an **order-n** method
  - $w_0 = \alpha$.
  - for $j = 0, 1, \cdots$,

$$w_{j+1} = w_j + h\,\phi\,(t_j, w_j, h)\,, \quad h = t_{j+1} - t_j,$$

  - **local truncation error** (LTE)

$$\tau_{j+1}(h) = \frac{y(t_{j+1}) - y(t_j)}{h} - \phi\,(t_j, y(t_j), h) = O\,(h^n)\,.$$

- Given tolerance $\tau > 0$, we would like to estimate **largest** step-size h for which

$$|\tau_{j+1}(h)| \lesssim \tau.$$

Approach: Estimate $\tau_{j+1}(h)$ with **order-**$(n+1)$ method

$$\widetilde{w}_{j+1} = \widetilde{w}_j + h\,\widetilde{\phi}\,(t_j, \widetilde{w}_j, h)\,, \quad \text{for} \quad j \geq 0.$$

LTE Estimation (I)

- Assume $\quad w_j \approx y(t_j), \quad \widetilde{w}_j \approx y(t_j)$ (only estimating **LTE**).

LTE Estimation (I)

- Assume $w_j \approx y(t_j)$, $\widetilde{w}_j \approx y(t_j)$ (only estimating **LTE**).
- $\phi(t, w, h)$ is order-n method

$$
\begin{aligned}
\tau_{j+1}(h) &\stackrel{def}{=} \frac{y(t_{j+1}) - y(t_j)}{h} - \phi\left(t_j, y(t_j), h\right) \\
&\stackrel{w_j \approx y(t_j)}{\approx} \frac{y(t_{j+1}) - \left(w_j + h\,\phi\left(t_j, w_j, h\right)\right)}{h} \\
&= \frac{y(t_{j+1}) - w_{j+1}}{h} = O\left(h^n\right).
\end{aligned}
$$

LTE Estimation (I)

- Assume $w_j \approx y(t_j)$, $\widetilde{w}_j \approx y(t_j)$ (only estimating **LTE**).
- $\phi(t, w, h)$ is order-n method

$$
\begin{aligned}
\tau_{j+1}(h) &\stackrel{def}{=} \frac{y(t_{j+1}) - y(t_j)}{h} - \phi\left(t_j, y(t_j), h\right) \\
&\stackrel{w_j \approx y(t_j)}{\approx} \frac{y(t_{j+1}) - \left(w_j + h\,\phi\left(t_j, w_j, h\right)\right)}{h} \\
&= \frac{y(t_{j+1}) - w_{j+1}}{h} = O\left(h^n\right).
\end{aligned}
$$

- $\widetilde{\phi}(t, w, h)$ is order-$(n+1)$ method,

$$
\begin{aligned}
\widetilde{\tau}_{j+1}(h) &\stackrel{def}{=} \frac{y(t_{j+1}) - y(t_j)}{h} - \widetilde{\phi}\left(t_j, y(t_j), h\right) \\
&\stackrel{\widetilde{w}_j \approx y(t_j)}{\approx} \frac{y(t_{j+1}) - \left(\widetilde{w}_j + h\,\widetilde{\phi}\left(t_j, \widetilde{w}_j, h\right)\right)}{h} \\
&= \frac{y(t_{j+1}) - \widetilde{w}_{j+1}}{h} = O\left(h^{n+1}\right).
\end{aligned}
$$

LTE Estimation (III)

- Assume $\quad w_j \approx y(t_j), \quad \widetilde{w}_j \approx y(t_j) \quad$ (only estimating **LTE**).

LTE Estimation (III)

- Assume $\quad w_j \approx y(t_j), \quad \widetilde{w}_j \approx y(t_j) \quad$ (only estimating **LTE**).
- $\phi(t, w, h)$ is order-n method

$$\tau_{j+1}(h) \quad \approx \quad \frac{y(t_{j+1}) - w_{j+1}}{h} = O\left(h^n\right).$$

LTE Estimation (III)

- Assume $w_j \approx y(t_j), \quad \widetilde{w}_j \approx y(t_j)$ (only estimating **LTE**).
- $\phi(t, w, h)$ is order-n method

$$\tau_{j+1}(h) \approx \frac{y(t_{j+1}) - w_{j+1}}{h} = O\left(h^n\right).$$

- $\widetilde{\phi}(t, w, h)$ is order-$(n+1)$ method,

$$\widetilde{\tau}_{j+1}(h) \approx \frac{y(t_{j+1}) - \widetilde{w}_{j+1}}{h} = O\left(h^{n+1}\right).$$

LTE Estimation (III)

- Assume $w_j \approx y(t_j)$, $\widetilde{w}_j \approx y(t_j)$ (only estimating **LTE**).
- $\phi(t, w, h)$ is order-n method

$$\tau_{j+1}(h) \approx \frac{y(t_{j+1}) - w_{j+1}}{h} = O\left(h^n\right).$$

- $\widetilde{\phi}(t, w, h)$ is order-$(n+1)$ method,

$$\widetilde{\tau}_{j+1}(h) \approx \frac{y(t_{j+1}) - \widetilde{w}_{j+1}}{h} = O\left(h^{n+1}\right).$$

- therefore

$$\begin{aligned}\tau_{j+1}(h) &\approx \frac{y(t_{j+1}) - \widetilde{w}_{j+1}}{h} + \frac{\widetilde{w}_{j+1} - w_{j+1}}{h} \\ &= O\left(h^{n+1}\right) + \frac{\widetilde{w}_{j+1} - w_{j+1}}{h} = O\left(h^n\right).\end{aligned}$$

LTE estimate: $\tau_{j+1}(h) \approx \dfrac{\widetilde{w}_{j+1} - w_{j+1}}{h}$

step-size selection (I)

$$\text{LTE estimate: } \tau_{j+1}(h) \approx \frac{\widetilde{w}_{j+1} - w_{j+1}}{h}$$

step-size selection (I)

$$\boxed{\text{LTE estimate: } \tau_{j+1}(h) \approx \frac{\widetilde{w}_{j+1} - w_{j+1}}{h}}$$

- Since $\tau_{j+1}(h) = O\left(h^n\right)$, **assume**

$$\tau_{j+1}(h) \approx K\, h^n \quad \text{where } K \text{ is independent of } h.$$

step-size selection (I)

$$\text{LTE estimate: } \tau_{j+1}(h) \approx \frac{\widetilde{w}_{j+1} - w_{j+1}}{h}$$

- Since $\tau_{j+1}(h) = O\left(h^n\right)$, **assume**

$$\tau_{j+1}(h) \approx K\, h^n \quad \text{where } K \text{ is independent of } h.$$

- K should satisfy

$$K\, h^n \approx \frac{\widetilde{w}_{j+1} - w_{j+1}}{h}.$$

- **Assume** LTE for new step-size $q\,h$ satisfies given tolerance ϵ:

$$\left|\tau_{j+1}(q\,h)\right| \leq \epsilon, \quad \text{need to estimate } q.$$

step-size selection (II)

$$\text{LTE estimate: } K\,h^n \approx \tau_{j+1}(h) \approx \frac{\widetilde{w}_{j+1} - w_{j+1}}{h}$$

step-size selection (II)

$$\text{LTE estimate: } K\,h^n \approx \tau_{j+1}(h) \approx \frac{\widetilde{w}_{j+1} - w_{j+1}}{h}$$

- **Assume** LTE for $q\,h$ satisfies given tolerance ϵ:

$$\begin{aligned} |\tau_{j+1}(q\,h)| &\approx |K\,(q\,h)^n| = q^n\,|K\,h^n| \\ &\approx q^n \left| \frac{\widetilde{w}_{j+1} - w_{j+1}}{h} \right| \leq \epsilon. \end{aligned}$$

step-size selection (II)

$$\text{LTE estimate: } K\,h^n \approx \tau_{j+1}(h) \approx \frac{\widetilde{w}_{j+1} - w_{j+1}}{h}$$

- **Assume** LTE for $q\,h$ satisfies given tolerance ϵ:

$$\begin{aligned} |\tau_{j+1}(q\,h)| &\approx |K\,(q\,h)^n| = q^n\,|K\,h^n| \\ &\approx q^n \left|\frac{\widetilde{w}_{j+1} - w_{j+1}}{h}\right| \leq \epsilon. \end{aligned}$$

$$\text{new step-size estimate: } q\,h \lesssim \left|\frac{\epsilon\,h}{\widetilde{w}_{j+1} - w_{j+1}}\right|^{\frac{1}{n}} h$$

Summary

- Given **order-**n method

$$w_{j+1} = w_j + h\,\phi\,(t_j, w_j, h)\,, \quad h = t_{j+1} - t_j, \quad j \geq 0,$$

- and given **order-**$(n+1)$ method

$$\widetilde{w}_{j+1} = \widetilde{w}_j + h\,\widetilde{\phi}\,(t_j, \widetilde{w}_j, h)\,, \quad \text{for} \quad j \geq 0.$$

- for each j, compute

$$\begin{aligned} w_{j+1} &= w_j + h\,\phi\,(t_j, w_j, h)\,, \\ \widetilde{w}_{j+1} &= w_j + h\,\widetilde{\phi}\,(t_j, w_j, h)\,, \end{aligned}$$

- new step-size $q\,h$ should satisfy

$$q\,h \lesssim \left|\frac{\epsilon\,h}{\widetilde{w}_{j+1} - w_{j+1}}\right|^{\frac{1}{n}} h = \left|\frac{\epsilon}{\widetilde{\phi}\,(t_j, w_j, h) - \phi\,(t_j, w_j, h)}\right|^{\frac{1}{n}} h.$$

Runge-Kutta-Fehlberg: 4^{th} order method, 5^{th} order estimate

$$\begin{aligned} w_{j+1} &= w_j + \frac{25}{216}k_1 + \frac{1408}{2565}k_3 + \frac{2197}{4104}k_4 - \frac{1}{5}k_5, \\ \widetilde{w}_{j+1} &= w_j + \frac{16}{135}k_1 + \frac{6656}{12825}k_3 + \frac{28561}{56430}k_4 - \frac{9}{50}k_5 + \frac{2}{55}k_6, \quad \text{where} \end{aligned}$$

Runge-Kutta-Fehlberg: 4^{th} order method, 5^{th} order estimate

$$
\begin{aligned}
w_{j+1} &= w_j + \frac{25}{216}k_1 + \frac{1408}{2565}k_3 + \frac{2197}{4104}k_4 - \frac{1}{5}k_5, \\
\widetilde{w}_{j+1} &= w_j + \frac{16}{135}k_1 + \frac{6656}{12825}k_3 + \frac{28561}{56430}k_4 - \frac{9}{50}k_5 + \frac{2}{55}k_6, \quad \text{where}
\end{aligned}
$$

$$
k_1 = h\, f\left(t_j, w_j\right),
$$

Runge-Kutta-Fehlberg: 4^{th} order method, 5^{th} order estimate

$$
\begin{aligned}
w_{j+1} &= w_j + \frac{25}{216}k_1 + \frac{1408}{2565}k_3 + \frac{2197}{4104}k_4 - \frac{1}{5}k_5, \\
\widetilde{w}_{j+1} &= w_j + \frac{16}{135}k_1 + \frac{6656}{12825}k_3 + \frac{28561}{56430}k_4 - \frac{9}{50}k_5 + \frac{2}{55}k_6, \quad \text{where}
\end{aligned}
$$

$$
\begin{aligned}
k_1 &= h\,f\left(t_j, w_j\right), \\
k_2 &= h\,f\left(t_j + \frac{h}{4}, w_j + \frac{1}{4}\,k_1\right), \\
k_3 &= h\,f\left(t_j + \frac{3h}{8}, w_j + \frac{3}{32}\,k_1 + \frac{9}{32}\,k_2\right), \\
k_4 &= h\,f\left(t_j + \frac{12h}{13}, w_j + \frac{1932}{2197}\,k_1 - \frac{7200}{2197}\,k_2 + \frac{7296}{2197}\,k_3\right), \\
k_5 &= h\,f\left(t_j + h, w_j + \frac{439}{216}\,k_1 - 8\,k_2 + \frac{3680}{513}\,k_3 - \frac{845}{4104}\,k_4\right), \\
k_6 &= h\,f\left(t_j + \frac{h}{2}, w_j - \frac{8}{27}\,k_1 + 2\,k_2 - \frac{3544}{2565}\,k_3 + \frac{1859}{4104}\,k_4 - \frac{11}{40}\,k_5\right).
\end{aligned}
$$

Runge-Kutta-Fehlberg: step-size selection procedure

- compute a conservative value for q:

$$q = \left| \frac{\epsilon\, h}{\mathbf{2}\left(\widetilde{w}_{j+1} - w_{j+1}\right)} \right|^{\frac{1}{4}}.$$

Runge-Kutta-Fehlberg: step-size selection procedure

- compute a conservative value for q:

$$q = \left| \frac{\epsilon\, h}{\mathbf{2}\left(\widetilde{w}_{j+1} - w_{j+1}\right)} \right|^{\frac{1}{4}}.$$

- make restricted step-size change:

$$h = \begin{cases} 0.1\, h, & \text{if } q \leq 0.1, \\ 4\, h, & \text{if } q \geq 4. \end{cases}$$

Runge-Kutta-Fehlberg: step-size selection procedure

- compute a conservative value for q:

$$q = \left| \frac{\epsilon\, h}{\mathbf{2}\left(\widetilde{w}_{j+1} - w_{j+1}\right)} \right|^{\frac{1}{4}}.$$

- make restricted step-size change:

$$h = \begin{cases} 0.1\, h, & \text{if } q \leq 0.1, \\ 4\, h, & \text{if } q \geq 4. \end{cases}$$

- step-size can't be too big:

$$h = \mathbf{min}\left(h, h_{\mathbf{max}}\right).$$

Runge-Kutta-Fehlberg: step-size selection procedure

- compute a conservative value for q:

$$q = \left| \frac{\epsilon\, h}{\mathbf{2}\left(\widetilde{w}_{j+1} - w_{j+1}\right)} \right|^{\frac{1}{4}}.$$

- make restricted step-size change:

$$h = \begin{cases} 0.1\, h, & \text{if } q \leq 0.1, \\ 4\, h, & \text{if } q \geq 4. \end{cases}$$

- step-size can't be too big:

$$h = \mathbf{min}\left(h, h_{\mathbf{max}}\right).$$

- step-size can't be too small:

$$\textbf{if} \quad h < h_{\mathbf{min}} \quad \textbf{then} \quad \textit{declare failure.}$$

Runge-Kutta-Fehlberg: example

Initial Value ODE $\quad \dfrac{dy}{dt} = y - t^2 + 1, \quad 0 \leq t \leq 2, \quad y(0) = 0.5,$

exact solution $\quad y(t) = (1 + t)^2 - 0.5\, e^t.$

Runge-Kutta-Fehlberg: example

Initial Value ODE $\frac{dy}{dt} = y - t^2 + 1, \quad 0 \leq t \leq 2, \quad y(0) = 0.5,$

exact solution $y(t) = (1+t)^2 - 0.5\, e^t.$

t_i	Exact	Euler $h = 0.025$	Modified Euler $h = 0.05$	Runge-Kutta Order Four $h = 0.1$
0.0	0.5000000	0.5000000	0.5000000	0.5000000
0.1	0.6574145	0.6554982	0.6573085	0.6574144
0.2	0.8292986	0.8253385	0.8290778	0.8292983
0.3	1.0150706	1.0089334	1.0147254	1.0150701
0.4	1.2140877	1.2056345	1.2136079	1.2140869
0.5	1.4256394	1.4147264	1.4250141	1.4256384

Runge-Kutta-Fehlberg: example

Initial Value ODE $\quad \dfrac{dy}{dt} = y - t^2 + 1, \quad 0 \le t \le 2, \quad y(0) = 0.5,$

exact solution $\quad y(t) = (1 + t)^2 - 0.5\, e^t.$

Runge-Kutta-Fehlberg: example

Initial Value ODE $\quad \frac{dy}{dt} = y - t^2 + 1, \quad 0 \le t \le 2, \quad y(0) = 0.5,$

exact solution $\quad y(t) = (1 + t)^2 - 0.5\, e^t.$

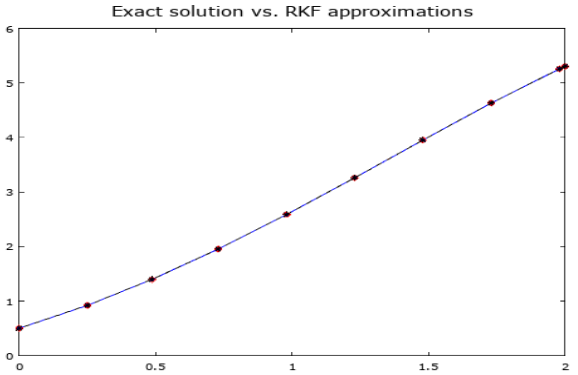
t_j	h_j	$y(t_j)$	w_j	$\widetilde{w}_j$
0.00000	0.00000	0.50000	0.50000	0.50000
0.25000	0.25000	0.92049	0.92049	0.92049
0.48655	0.23655	1.39649	1.39649	1.39649
0.72933	0.24278	1.95374	1.95375	1.95375
0.97933	0.25000	2.58642	2.58643	2.58643
1.22933	0.25000	3.26045	3.26046	3.26046
1.47933	0.25000	3.95208	3.95210	3.95210
1.72933	0.25000	4.63081	4.63083	4.63083
1.97933	0.25000	5.25747	5.25749	5.25749
2.00000	0.02067	5.30547	5.30549	5.30549

Runge-Kutta-Fehlberg: solution plots

Initial Value ODE $$\frac{dy}{dt} = y - t^2 + 1, \quad 0 \le t \le 2, \quad y(0) = 0.5.$$

Runge-Kutta-Fehlberg: solution plots

Initial Value ODE $$\frac{dy}{dt} = y - t^2 + 1, \quad 0 \le t \le 2, \quad y(0) = 0.5.$$



Runge-Kutta-Fehlberg: solution errors

Initial Value ODE $$\frac{dy}{dt} = y - t^2 + 1, \quad 0 \leq t \leq 2, \quad y(0) = 0.5.$$

Runge-Kutta-Fehlberg: solution errors

Initial Value ODE $\quad \dfrac{dy}{dt} = y - t^2 + 1, \quad 0 \le t \le 2, \quad y(0) = 0.5.$

Runge-Kutta-Fehlberg: truncation errors

Initial Value ODE $$\frac{dy}{dt} = y - t^2 + 1, \quad 0 \le t \le 2, \quad y(0) = 0.5.$$

$$\textbf{actual} \stackrel{def}{=} \left| \frac{y(t_j) - w_j}{h_j} \right|, \quad \textbf{estimate} \stackrel{def}{=} \left| \frac{\widetilde{w}_j - w_j}{h_j} \right|.$$

Runge-Kutta-Fehlberg: truncation errors

Initial Value ODE $$\frac{dy}{dt} = y - t^2 + 1, \quad 0 \le t \le 2, \quad y(0) = 0.5.$$

$$\textbf{actual} \stackrel{def}{=} \left| \frac{y(t_j) - w_j}{h_j} \right|, \quad \textbf{estimate} \stackrel{def}{=} \left| \frac{\widetilde{w}_j - w_j}{h_j} \right|.$$