Practical introduction to time integration

The concept:

ODE $\underbrace{\frac{du}{dt} = f(u)}$ initial condition $u(t=0) = u_0$

"right hand side" RHS

Solution : function $u(t)$

on the computer we will get discrete $u(n\Delta t)$

u_0, u_1, u_2, ..., u_n, ...

$t=0$, $t=\Delta t$, $t=2\Delta t$, ..., $t=n\Delta t$, ...

on the computer we know u_0 because it is the initial condition

3 Runge–Kutta Method

We will use the standard third-order Runge–Kutta method (the same method we discussed in class) to numerically integrate a couple of ODEs. To advance the solution u from time t to $t + \Delta t$, three sub-steps, are taken. If the solution at time t is u_n the following three steps are taken to advance the solution to u_{n+1} at $t + \Delta t$:

$$u_{n+\alpha} = u_n + \Delta t \frac{1}{2} f(u_n) \qquad \text{Substep 1} \tag{3}$$

$$u_{n+\alpha_1} = u_n + \Delta t \left(-f(u_n) + 2f(u_{n+\alpha})\right) \qquad \text{Substep 2} \tag{4}$$

$$u_{n+1} = u_n + \Delta t \left(\frac{1}{6} f(u_n) + \frac{2}{3} f(u_{n+\alpha}) + \frac{1}{6} f(u_{n+\alpha_1})\right) \qquad \text{Substep 3} \tag{5}$$

$t=0$ $\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad$ $t=\Delta t$

$$u_0 \xrightarrow[\text{use eq. 3}]{\text{substep 1}} u_{0+\alpha} \xrightarrow[\text{use eq. 4}]{\text{substep 2}} u_{0+\alpha_1} \xrightarrow[\text{use eq. 5}]{\text{substep 3}} u_1$$

repeat

$t=\Delta t$ $\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad$ $t=2\Delta t$

$$u_1 \xrightarrow[\text{use eq. 3}]{} u_{1+\alpha} \xrightarrow[\text{use eq. 4}]{} u_{1+\alpha_1} \xrightarrow[\text{use eq. 5}]{} u_2$$

repeat

$t=2\Delta t$ $\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad$ $t=3\Delta t$

$$u_2 \xrightarrow[\text{use eq. 3}]{} u_{2+\alpha} \xrightarrow[\text{use eq. 4}]{} u_{2+\alpha_1} \xrightarrow[\text{use eq. 5}]{} u_3$$

$\vdots$

keep doing this until you reach the end time

```
function [u, time]=rkexample(dt, tend)

time=0:dt:tend;
np=length(time);

u=zeros(size(time));
u(1)=1;
```

space for variables

initial condition

```
for n=1:np-1
    r0=rhs(u(n));
    u1=update(dt, [1/2 0 0], u(n), r0, 0, 0);
```

$f(u_n)$

$u_{n+\alpha}$

substep 1

```
    r1=rhs(u1);
    u2=update(dt, [-1 2 0], u(n), r0, r1, 0);
```

$f(u_{n+\alpha})$

$u_{n+\alpha_2}$

substep 2

```
    r2=rhs(u2);
    u(n+1)=update(dt, [1/6 2/3 1/6], u(n), r0, r1, r2);
end
```

$f(u_{n+\alpha_2})$

u_{n+1}

substep 3

```
    function u=update(dt, a, u, rhs1, rhs2, rhs3)
        u=u+dt*(a(1)*rhs1+a(2)*rhs2+a(3)*rhs3);
    end
```

RK substep:

$$u_{n+\bullet} = u_n + \Delta t\left(a f(u_1) + b f(u_2) + c f(u_3)\right)$$

```
    function r=rhs(u)
        r=u;
    end


end
```

$f(u)$