Alexander Knebe, *Universidad Autonoma de Madrid*

Ode to Joy

(**O**rdinary **d**ifferential **e**quations)

L.V.Beethoven

Numerical (Astro-)Physics:

solving differential equations
of the physics under investigation
using computers

- **N-body**
  - plasma physics
  - molecular dynamics
  - stellar systems, e.g. globular clusters
  - individual galaxies
  - the whole Universe

- **Fluid Dynamics**
  - solar and stellar physics
  - magneto-hydrodynamics
  - aero-plane design
  - meteorology
  - oceanography

- **N-body**
  - plasma physics
  - molecular dynamics
  - stellar systems, e.g. globular clusters
  - individual galaxies
  - the whole Universe

- **Fluid Dynamics**
  - solar and stellar physics
  - magneto-hydrodynamics
  - aero-plane design
  - meteorology
  - oceanography

- **N-body**
  - plasma physics
  - molecular dynamics
  - stellar systems, e.g. globular clusters
  - individual galaxies
  - **the whole Universe**

- **Fluid Dynamics**
  - solar and stellar physics
  - magneto-hydrodynamics
  - aero-plane design
  - meteorology
  - oceanography

The N-body Problem

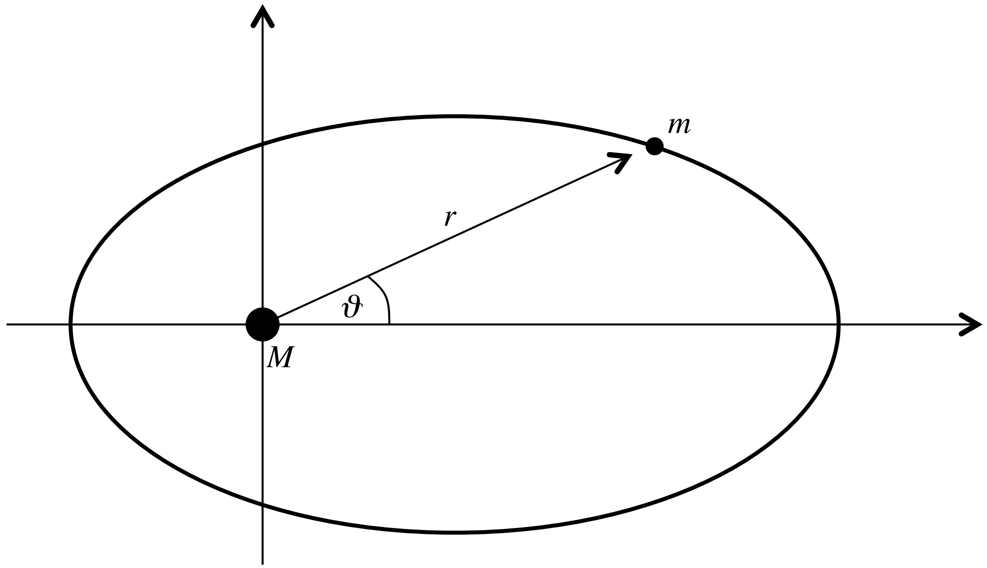
- the equations
  - Newton's second law of motion

$$m\frac{d^2r}{dt^2} = F(r)$$

  - Newton's law of gravity

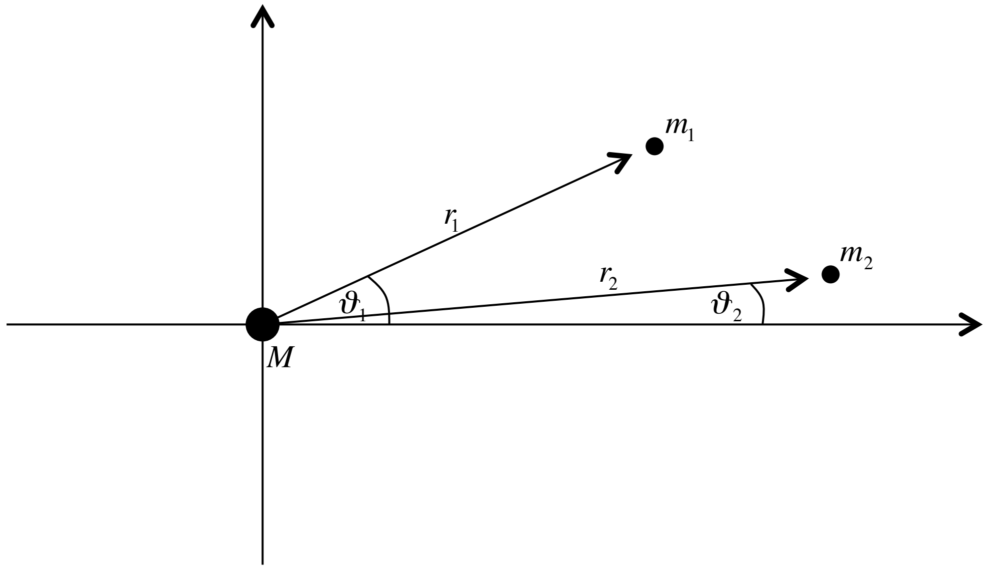
$$F = G\frac{Mm}{r^2}$$

- the two-body problem



$$r(\vartheta) = \frac{k}{1 + \varepsilon \cos(\vartheta)}$$

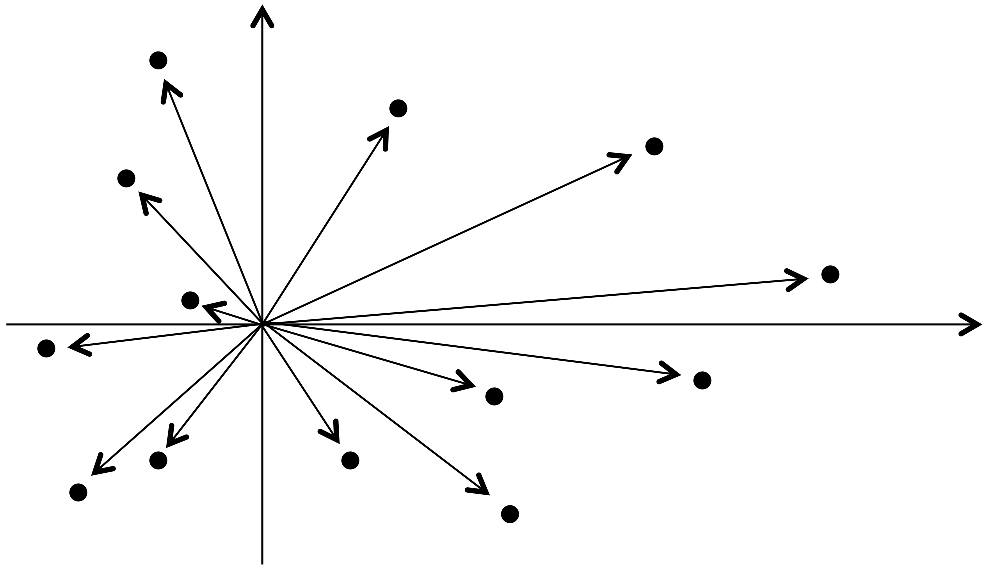
- the three-body problem



no analytical solution!

numerical integration required...

- the N-body problem



objective of this lecture series!

Numerical Integration of Ordinary Differential Equations

- ordinary differential equation

$$0 = G(f^{(n)}, f^{(n-1)}, \ldots, f^{(2)}, f^{(1)}, f^{(0)}, t) \qquad f^{(n)} \equiv \frac{d^n f}{dt^n}$$

- ordinary differential equation

$$0 = G(f^{(n)}, f^{(n-1)}, \ldots, f^{(2)}, f^{(1)}, f^{(0)}, t) \qquad f^{(n)} \equiv \frac{d^n f}{dt^n}$$

Note the change to *t* as independent variable,

but in physics we are mostly concerned with the temporal evolution of processes...

- 1^{st} order explicit ordinary differential equation

$$\frac{df}{dt} = G(f, t)$$

- 1st order explicit ordinary differential equation

$$\frac{df}{dt} = G(f,t)$$

$$\Rightarrow \quad \frac{\Delta f}{\Delta t} = \frac{f(t_{i+1}) - f(t_i)}{t_{i+1} - t_i} = \frac{f_{i+1} - f_i}{t_{i+1} - t_i} = G(f_i, t_i)$$

$$\Rightarrow \quad f_{i+1} = f_i + \Delta t \, G(f_i, t_i)$$

- 1st order explicit ordinary differential equation

$$\frac{df}{dt} = G(f,t)$$

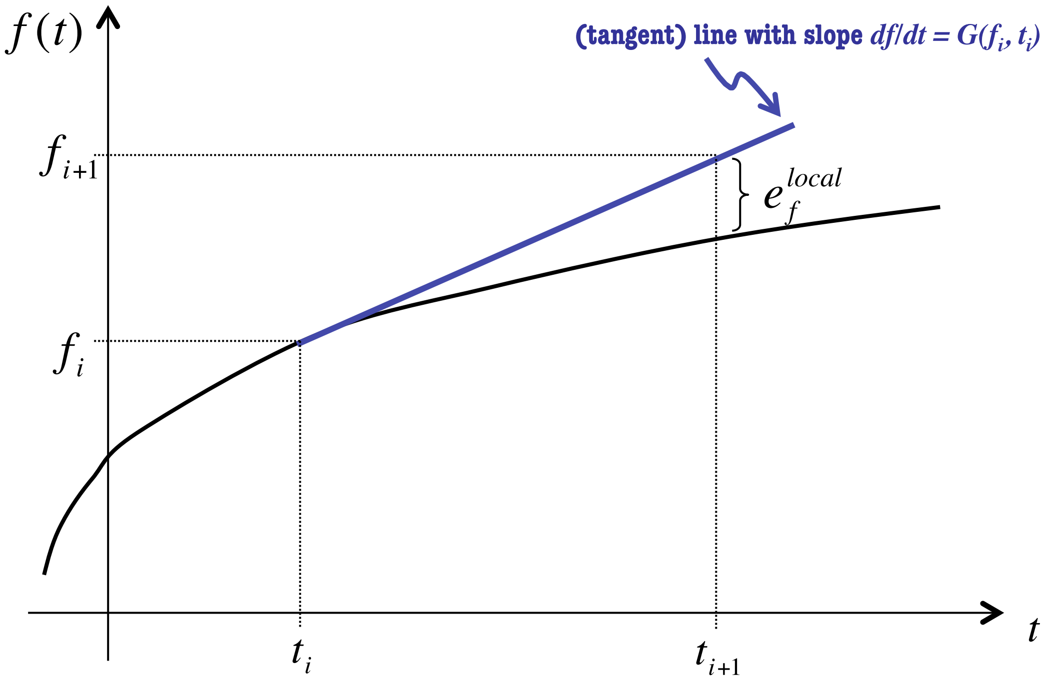
$$=> \quad \frac{\Delta f}{\Delta t} = \frac{f(t_{i+1}) - f(t_i)}{t_{i+1} - t_i} = \frac{f_{i+1} - f_i}{t_{i+1} - t_i} = G(f_i, t_i)$$

$$=> \quad f_{i+1} = f_i + \Delta t\, G(f_i, t_i)$$

Taylor expansion of $f(x)$ about x_i up to 1st order...

$$f(t_i + \Delta t) = f(t_i) + f^{(1)}(t_i)\, dt$$

- 1^{st} order explicit ordinary differential equation
  - Euler scheme



$$\Rightarrow \quad f_{i+1} = f_i + \Delta t \, G(f_i, t_i)$$

- 1^{st} order explicit ordinary differential equation
    - Euler scheme

$$f_{i+1} = f_i + \Delta t \, G(f_i, t_i)$$

$$\frac{df}{dt} = G(f,t)$$

first term in Taylor expansion of $f(t)$ about t_i !

- 1st order explicit ordinary differential equation

  - Euler scheme

$$f_{i+1} = f_i + \Delta t \; G(f_i, t_i)$$

  - local error estimate (F = correct solution, f = numerical solution)

$$F(t_{i+1}) = F(t_i) + \Delta t \; \dot{F}(t_i) + \frac{(\Delta t)^2}{2} \ddot{F}(t_i) + \ldots$$

$$e_f^{local} = F(t_{i+1}) - f_{i+1} = F(t_i) + \Delta t \; \dot{F}(t_i) + \frac{(\Delta t)^2}{2} \ddot{F}(t_i) + \ldots - \left(f_i + \Delta t \; G(f_i, t_i) \right)$$

=> $$e_f^{local} \propto (\Delta t)^2$$

- 1st order explicit ordinary differential equation

  - Euler scheme

$$f_{i+1} = f_i + \Delta t\ G(f_i, t_i)$$

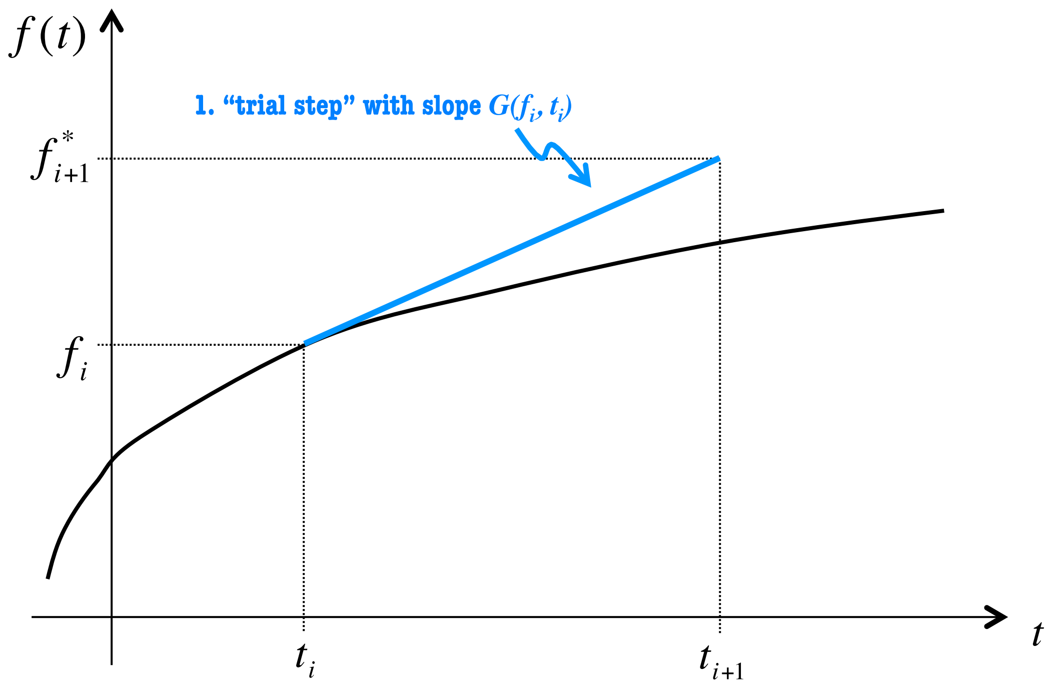
  - global error estimate

$$e_f^{global} \propto \sum_{i=1}^{N} (\Delta t)^2 = N(\Delta t)^2 = \frac{T_N - T_0}{\Delta t}(\Delta t)^2 \propto \Delta t$$

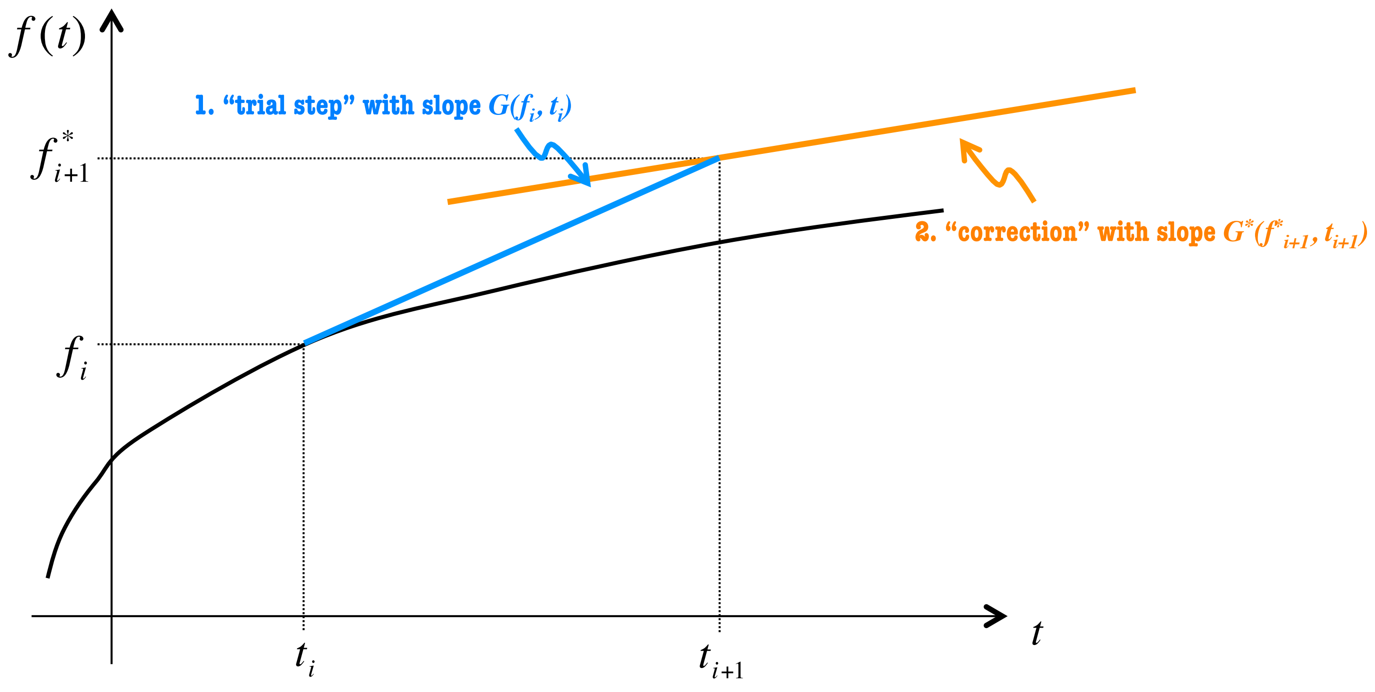
=> $e_f^{global} \propto (\Delta t)$ "first order accurate"

increase accuracy by including higher derivates...

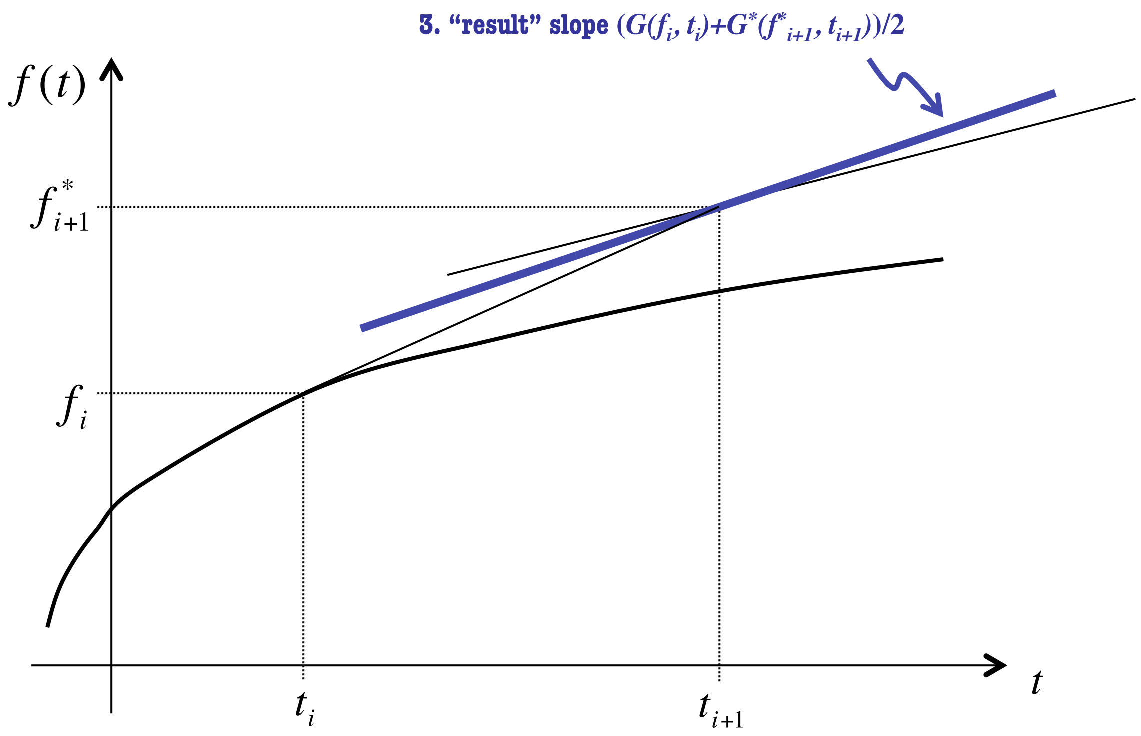
- 1^{st} order explicit ordinary differential equation
  - modified Euler scheme



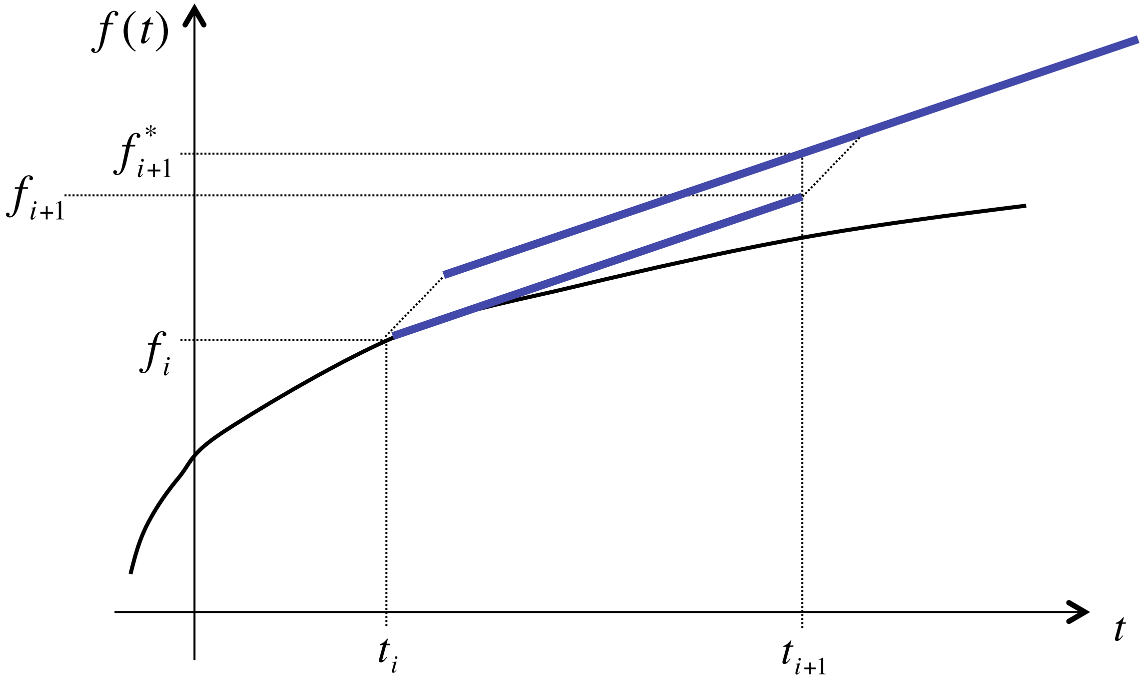
- 1^{st} order explicit ordinary differential equation
    - modified Euler scheme



- 1st order explicit ordinary differential equation
  - modified Euler scheme

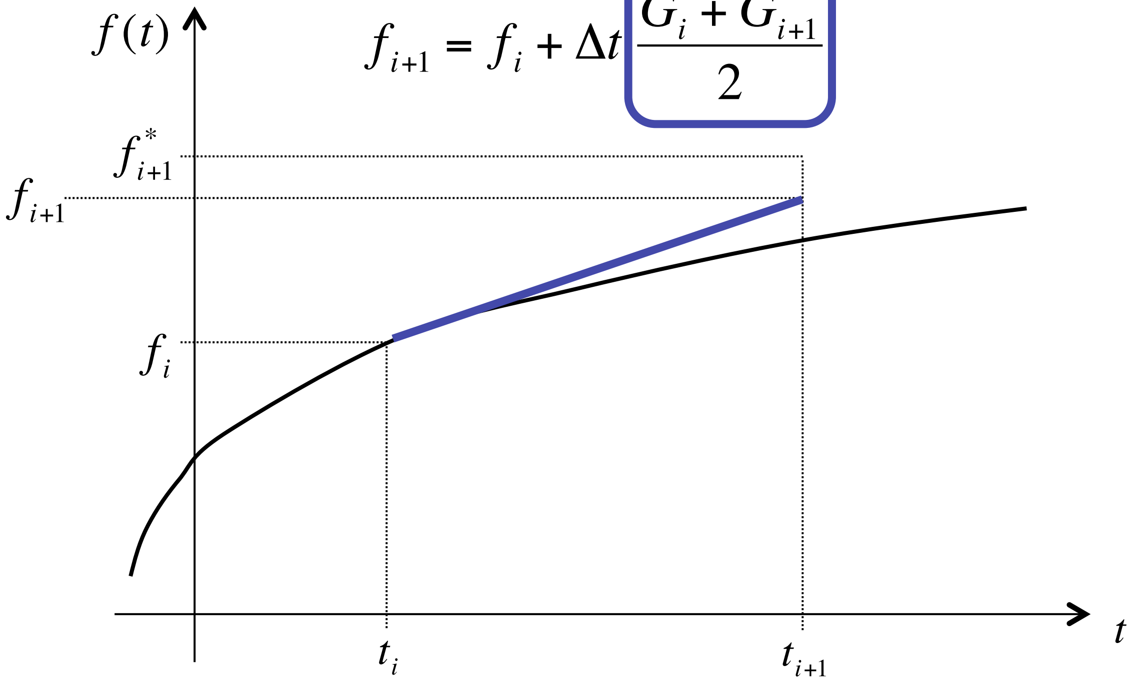


- 1^{st} order explicit ordinary differential equation
    - modified Euler scheme



- 1^{st} order explicit ordinary differential equation
    - modified Euler scheme

$$f_{i+1} = f_i + \Delta t \frac{G_i + G_{i+1}^*}{2}$$

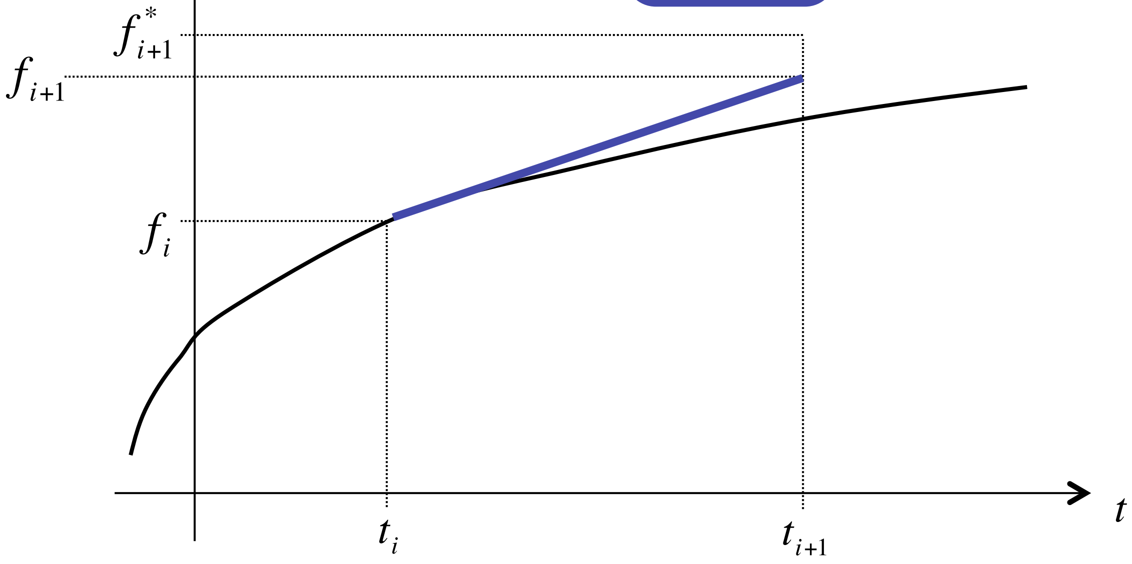


- 1^{st} order explicit ordinary differential equation
  - modified Euler scheme

$$f_{i+1} = f_i + \Delta t \frac{G_i + G^*_{i+1}}{2}$$

additional calculations required!

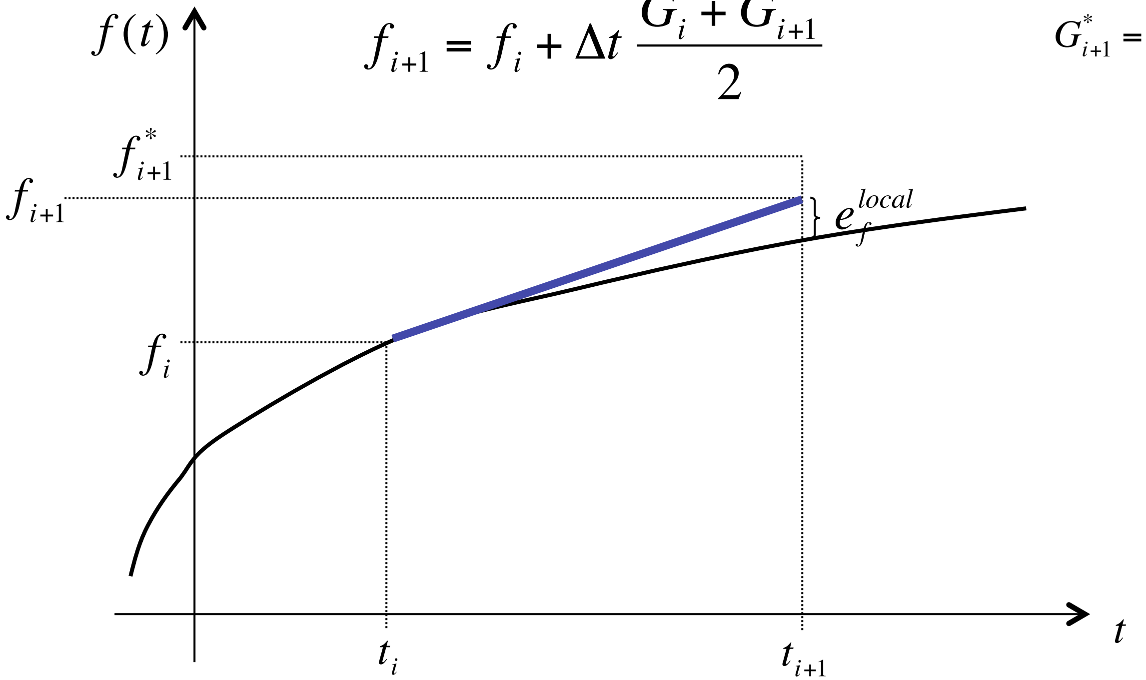
$$G^*_{i+1} = G(t_i + \Delta t, f_i + \Delta t\, G_i)$$



- 1st order explicit ordinary differential equation
  - modified Euler scheme

$$f_{i+1} = f_i + \Delta t \, \frac{G_i + G^*_{i+1}}{2}$$

$$G^*_{i+1} = G(t_i + \Delta t, f_i + \Delta t \, G_i)$$



- 1^{st} order explicit ordinary differential equation
  - modified Euler scheme

$$f_{i+1} = f_i + \Delta t \, \frac{G_i + G^*_{i+1}}{2}$$

$$G^*_{i+1} = G(t_i + \Delta t, f_i + \Delta t \, G_i)$$

  - local error estimate

$$\dot{G}_i = \frac{G^*_{i+1} - G_i}{\Delta t}$$

- 1st order explicit ordinary differential equation

  - modified Euler scheme

$$f_{i+1} = f_i + \Delta t \, \frac{G_i + G^*_{i+1}}{2} \qquad G^*_{i+1} = G(t_i + \Delta t, f_i + \Delta t \, G_i)$$

  - local error estimate

$$\dot{G}_i = \frac{G^*_{i+1} - G_i}{\Delta t} \quad \Rightarrow \quad G^*_{i+1} = \Delta t \dot{G}_i + G_i$$

- 1^{st} order explicit ordinary differential equation
  - modified Euler scheme

$$f_{i+1} = f_i + \Delta t \, \frac{G_i + G_{i+1}^*}{2} \qquad G_{i+1}^* = G(t_i + \Delta t, f_i + \Delta t \, G_i)$$

  - local error estimate

$$\dot{G}_i = \frac{G_{i+1}^* - G_i}{\Delta t}$$

$$f_{i+1} = f_i + \Delta t \; G_i + \frac{1}{2}(\Delta t)^2 \dot{G}_i + \ldots$$

additional term!

- 1^{st} order explicit ordinary differential equation
  - modified Euler scheme

$$f_{i+1} = f_i + \Delta t \, \frac{G_i + G_{i+1}^*}{2} \qquad G_{i+1}^* = G(t_i + \Delta t, f_i + \Delta t \, G_i)$$

  - local error estimate

$$\dot{G}_i = \frac{G_{i+1}^* - G_i}{\Delta t}$$

$$f_{i+1} = f_i + \Delta t \; G_i + \frac{1}{2}(\Delta t)^2 \dot{G}_i + \ldots$$

additional term!

comparison to F_{i+1} => $e_f^{local} \propto (\Delta t)^3$

- 1^{st} order explicit ordinary differential equation
  - modified Euler scheme

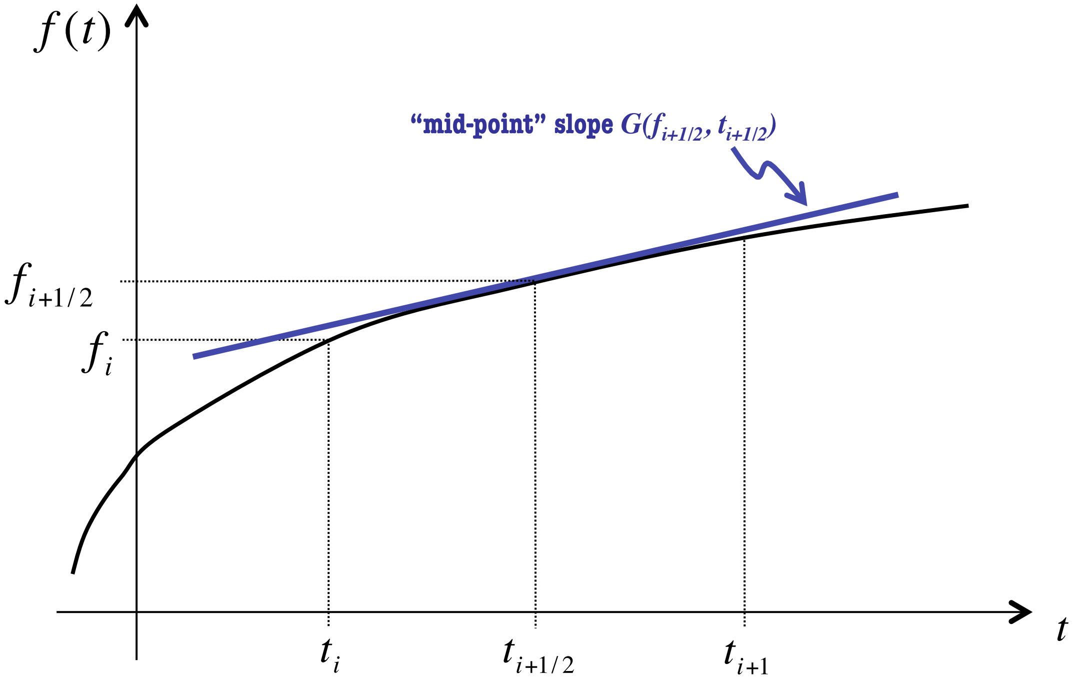
$$f_{i+1} = f_i + \Delta t \, \frac{G_i + G^*_{i+1}}{2} \qquad G^*_{i+1} = G(t_i + \Delta t, f_i + \Delta t \, G_i)$$

  - global error estimate

…

=> $e_f^{global} \propto (\Delta t)^2$ "second order accurate"
(at the expanse of more calculations)

- 1^{st} order explicit ordinary differential equation
  - 2^{nd} order Runge-Kutta scheme

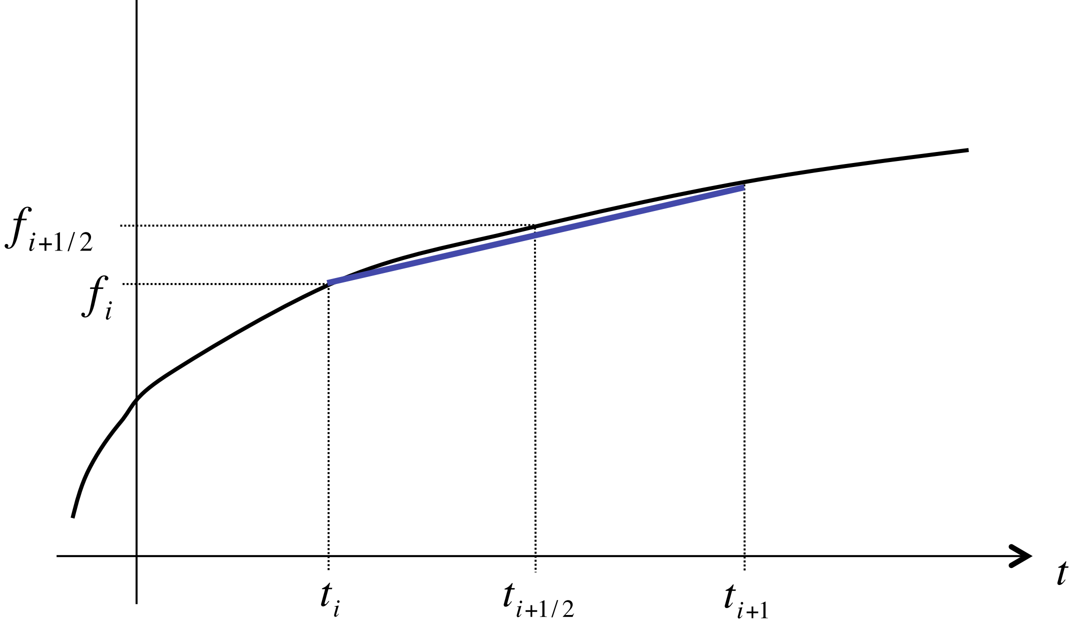


- 1st order explicit ordinary differential equation
  - 2nd order Runge-Kutta scheme

$$f_{i+1} = f_i + \Delta t \, G(f_{1+1/2}, t_{i+1/2})$$

additional calculations required!

$$t_{1/2} = t_i + \Delta t/2$$
$$f_{1/2} = f_i + G(f_i, t_i)\, \Delta t/2$$



- 1st order explicit ordinary differential equation
  - 2nd order Runge-Kutta scheme

$$f_{i+1} = f_i + \Delta t\, G(f_{1+1/2}, t_{i+1/2})$$

$$t_{1/2} = t_i + \Delta t/2$$
$$f_{1/2} = f_i + G(f_i, t_i)\,\Delta t/2$$

  - global error estimate

…

modified Euler & 2nd order Runge-Kutta agree to 1st order (cf. I_3 and I_4 in Numerical Integration)

=> $e_f^{global} \propto (\Delta t)^2$ "second order accurate"
(at the expanse of more calculations)

- 1^{st} order explicit ordinary differential equation
  - 4^{th} order Runge-Kutta scheme

$$f_{i+1} = f_i + \Delta t \; \frac{1}{6}(k_1 + 2k_2 + 2k_3 + k_4)$$

$$k_1 = G\left(f_i, t_i\right)$$
$$k_2 = G\left(f_i + k_1 \frac{(t_{i+1} - t_i)}{2}, t_i + \frac{(t_{i+1} - t_i)}{2}\right)$$
$$k_3 = G\left(f_i + k_2 \frac{(t_{i+1} - t_i)}{2}, t_i + \frac{(t_{i+1} - t_i)}{2}\right)$$
$$k_4 = G\left(f_i + k_3 (t_{i+1} - t_i), t_{i+1}\right)$$

  - global error estimate

=> $e_f^{global} \propto (\Delta t)^4$ "fourth order accurate"

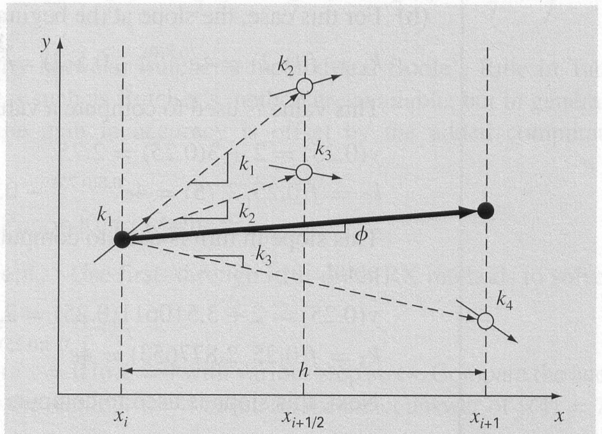
(at the expanse of *far* more calculations)

- 1^{st} order explicit ordinary differential equation

  - 4^{th} order Runge-Kutta scheme

$$f_{i+1} = f_i + \Delta t \; \frac{1}{6}(k_1 + 2k_2 + 2k_3 + k_4)$$

Graphical depiction of the slope estimates comprising the fourth-order RK method.



$$k_1 = G\left(f_i, t_i\right)$$

$$k_2 = G\left(f_i + k_1 \frac{(t_{i+1} - t_i)}{2}, t_i + \frac{(t_{i+1} - t_i)}{2}\right)$$

$$k_3 = G\left(f_i + k_2 \frac{(t_{i+1} - t_i)}{2}, t_i + \frac{(t_{i+1} - t_i)}{2}\right)$$

$$k_4 = G\left(f_i + k_3(t_{i+1} - t_i), t_{i+1}\right)$$

  - global error estimate

=> $e_f^{global} \propto (\Delta t)^4$ "fourth order accurate"

(at the expanse of *far* more calculations)

- 2nd order explicit ordinary differential equation

$$\frac{d^2 f}{dt^2} = G(f^{(1)}, f, t)$$

- 2nd order explicit ordinary differential equation

$$\frac{df}{dt} = h(f,h,t)$$
$$\frac{dh}{dt} = g(f,h,t)$$

single 2nd order equation →

← *system of two 1st order equations*

$$\frac{d^2 f}{dt^2} = G(f^{(1)}, f, t)$$

- 2^{nd} order explicit ordinary differential equation

$$\frac{df}{dt} = h(f,h,t)$$
$$\frac{dh}{dt} = g(f,h,t)$$

single 2^{nd} order equation →

← *system of two 1^{st} order equations*

$$\frac{d^2 f}{dt^2} = G(f^{(1)}, f, t)$$

solve using schemes for 1^{st} order equations

- 2^{nd} order explicit ordinary differential equation

$$\frac{df}{dt} = h(f,h,t)$$
$$\frac{dh}{dt} = g(f,h,t)$$

single 2^{nd} order equation →

← system of two 1^{st} order equations

$$\frac{d^2 f}{dt^2} = G(f^{(1)},f,t)$$

solve using schemes for 1^{st} order equations:

- Euler method

$$f_{i+1} = f_i + h(f_i,h_i,t_i)\ \Delta t$$
$$h_{i+1} = h_i + g(f_i,h_i,t_i)\ \Delta t$$

- 2^{nd} order Runge-Kutta

$$t_{mid} = t_i + \Delta t/2$$
$$f_{mid} = f_i + h(f_i,h_i,t_i)\ \Delta t/2$$
$$h_{mid} = h_i + g(f_i,h_i,t_i)\ \Delta t/2$$

$$f_{i+1} = f_i + h(f_{mid},h_{mid},t_{mid})\ \Delta t$$
$$h_{i+1} = h_i + g(f_{mid},h_{mid},t_{mid})\ \Delta t$$

- 2nd order explicit ordinary differential equation

$$\frac{df}{dt} = h(f,h,t)$$
$$\frac{dh}{dt} = g(f,h,t)$$

single 2nd order equation →

← *system of two 1st order equations*

$$\frac{d^2 f}{dt^2} = G(f^{(1)}, f, t)$$

solve using schemes for 1st order equations:

- Euler method

$$f_{i+1} = f_i + h(f_i,h_i,t_i)\,\Delta t$$
$$h_{i+1} = h_i + g(f_i,h_i,t_i)\,\Delta t$$

- 2nd order Runge-Kutta

$$t_{mid} = t_i + \Delta t/2$$
$$f_{mid} = f_i + h(f_i,h_i,t_i)\,\Delta t/2$$
$$h_{mid} = h_i + g(f_i,h_i,t_i)\,\Delta t/2$$

arbitrary functions of f,h,t

$$f_{i+1} = f_i + h(f_{mid},h_{mid},t_{mid})\,\Delta t$$
$$h_{i+1} = h_i + g(f_{mid},h_{mid},t_{mid})\,\Delta t$$

- 2^{nd} order explicit ordinary differential equation
  - leap-frog scheme

$$\frac{d^2 f}{dt^2} = G(f)$$

- 2^{nd} order explicit ordinary differential equation
  - leap-frog scheme

$$\frac{df}{dt} = h\,,\ \frac{dh}{dt} = G(f)$$

$$f_{i+1} = f_i + \Delta t\, h_{i+1/2}$$

$$h_{i+3/2} = h_{i+1/2} + \Delta t\, G(f_{i+1})$$

- 2^{nd} order explicit ordinary differential equation
  - leap-frog scheme

$$\frac{df}{dt} = h\,, \quad \frac{dh}{dt} = G(f)$$

df/dt **only depends on** *h*
dh/dt **only depends on** *f*

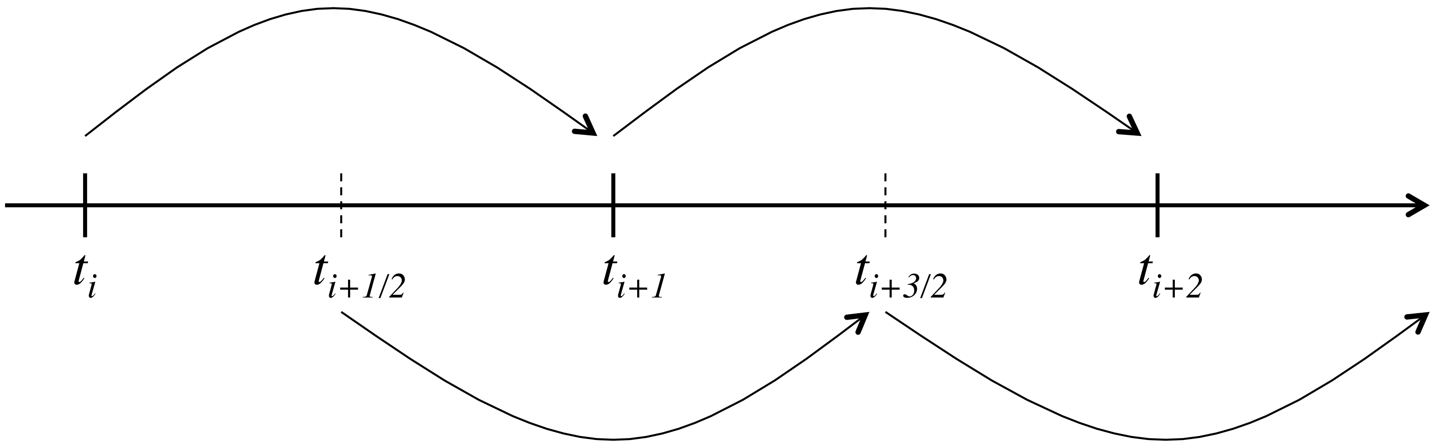
$$f_{i+1} = f_i + \Delta t\, h_{i+1/2}$$

$$h_{i+3/2} = h_{i+1/2} + \Delta t\, G(f_{i+1})$$

- 2^{nd} order explicit ordinary differential equation
  - leap-frog scheme

$$\frac{d^2 f}{dt^2} = G(f)$$

$$f_{i+1} = f_i + \Delta t \, h_{i+1/2}$$



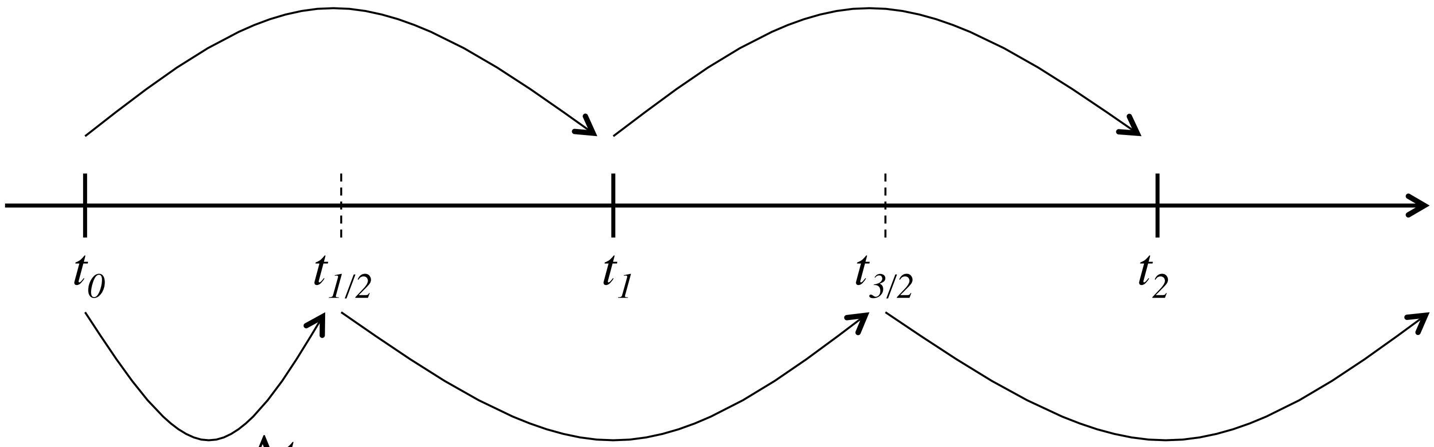
$$h_{i+3/2} = h_{i+1/2} + \Delta t \, G(f_{i+1})$$

- 2^{nd} order explicit ordinary differential equation

    - leap-frog scheme - jumpstart

$$\frac{d^2 f}{dt^2} = G(f)$$

$$f_1 = f_0 + \Delta t \, h_{1/2}$$

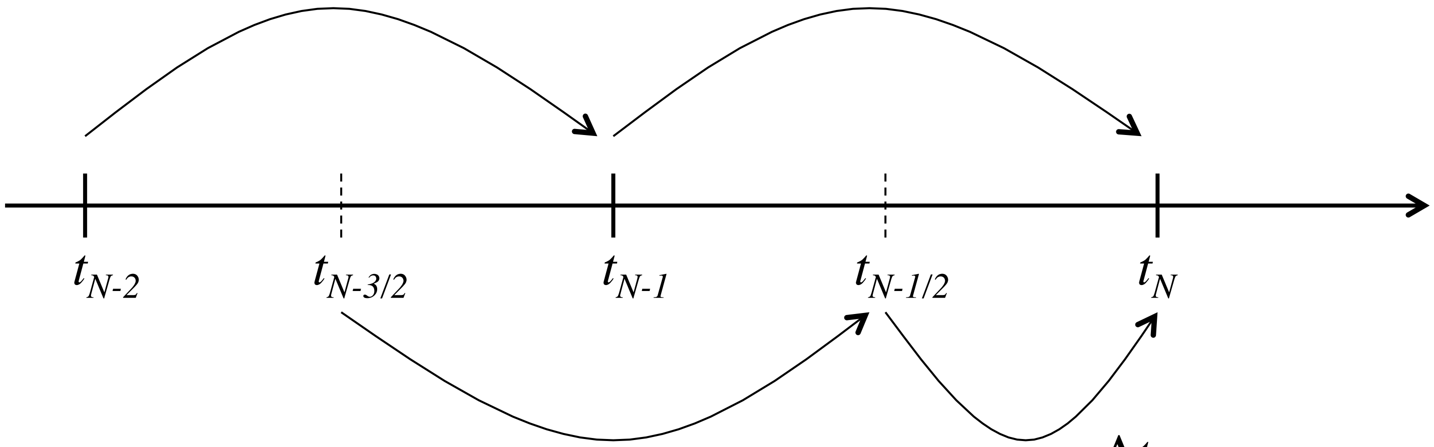


$$h_{1/2} = h_0 + \frac{\Delta t}{2} G(f_0)$$

- 2nd order explicit ordinary differential equation
  - leap-frog scheme - resync

$$\frac{d^2 f}{dt^2} = G(f)$$

$$f_N = f_{N-1} + \Delta t\, h_{N-1/2}$$



$$h_N = h_{N-1/2} + \frac{\Delta t}{2} G(f_N)$$

- 2nd order explicit ordinary differential equation
  - leap-frog scheme

$$\frac{d^2 f}{dt^2} = G(f) \longrightarrow \begin{aligned} f_{i+1} &= f_i + \Delta t\, h_{i+1/2} \\ h_{i+3/2} &= h_{i+1/2} + \Delta t\, G(f_{i+1}) \end{aligned}$$

+ second order accurate scheme

+ no additional calculations

+ symplectic scheme (energy conservation...)

+ very well suited for systems of type $\frac{d^2 f}{dt^2} = G(f)$

"Numerical Gravity"

- the equations

  - Newton's second law of motion

$$m\frac{d^2r}{dt^2} = F(r)$$

  - Newton's law of gravity

$$F = G\frac{Mm}{r^2}$$

- the equations

  - Newton's second law of motion

$$\frac{dr}{dt} = v, \quad \frac{dv}{dt} = f(r)$$

  - Newton's law of gravity

$$f(r) = G\frac{M}{r^2}$$

leap-frog integrator!

- leap-frog integration

$$m\frac{d^2r}{dt^2} = F(r)$$

$$F(r) = G\frac{Mm}{r^2}$$

$$\Longrightarrow$$

$$r_{i+1} = r_i + \Delta t\, v_{i+1/2}$$

$$v_{i+3/2} = v_{i+1/2} + \Delta t\, f(r_i)$$

$$f(r_i) = G\frac{M}{r_i^2}$$

dawn of N-body simulations