

Ordinary differential equations (part 2)

Outline:

- Runge-Kutta methods (1-st order ODE)
 - derivation of RK2 methods (Heun and midpoint)
 - general formulation of RK methods – Butcher's tableau
 - classical RK4 method
 - example: simple harmonic oscillator (2-nd order ODE) – energy drift
- symplectic integrators (1-st order ODE)

- The basic Euler methods are only the 1-st order methods, and they provide the low accuracy solutions. Trapezoid method is 2-nd order, but due to implicit character some additional work must be done in each time step (subiteration) and the order is still not high.
- There are another families of methods of higher order allowing to solve ODE numerically:
 - **one-step**: Runge-Kutta methods (first invented at the turn 19/20 centuries – no computers!)
Runge(1895), Heun(1900), Kutta (1901)-generalization
 - **multistep**: Adams-Bashforth, Adams-Moulton and backward differentiation methods
(some invented in mid- and at the end of 19 century – no computers!)
Adams (1845) – first (?) attempt to predict the orbit of unknown planet (Neputne),
Bashforth(1883), Moulton (1926)
 - symplectic integrators (designed for solving the special kind of problems - defined in phase-space)

RK2 method

The Taylor series for $u(t+\Delta t)$ reads

$$u(t + \Delta t) = u(t) + \underbrace{\frac{du}{dt} \Delta t}_{\text{Euler part}} + \frac{1}{2} \frac{d^2 u}{dt^2} \Delta t^2 + O(\Delta^3)$$

now we aim at higher order formula
new local truncation error

let's find the expression for the second derivative

$$\frac{du}{dt} = f$$

$$\frac{d^2 u}{dt^2} = \frac{d}{dt} \frac{du}{dt} = \frac{d}{dt} f = \frac{\partial f}{\partial t} + \frac{\partial f}{\partial u} \frac{du}{dt} = f_t + f_u f$$

- assumed abbreviations

$$\frac{du}{dt} \equiv u_t, \quad \frac{\partial f}{\partial t} \equiv f_t, \quad \frac{\partial f}{\partial u} \equiv f_u$$

now rewrite the Taylor series for $u(t+\Delta t)$

$$\begin{aligned} u(t + \Delta t) &= u(t) + \Delta t f + \frac{\Delta t^2}{2} (f_t + f f_u) + O(\Delta t^3) \\ &= u(t) + \frac{\Delta t}{2} f + \frac{\Delta t}{2} \underbrace{(f + \Delta t f_t + \Delta t f f_u)}_{?} + O(\Delta t^3) \end{aligned}$$

to find the alternative expression for (?) we again use the Taylor expansion but for the term $f(t+\Delta t, u+\Delta t f)$

$$f(t + \Delta t, u + \Delta t f) = f + \Delta t (f_t + f_u f) + O(\Delta t^2)$$

now we may replace (?) with this expansion, then we get new formula but with the same 3-rd order accuracy

$$u(t + \Delta t) = u(t) + \frac{\Delta t}{2} [f(t, u) + f(t + \Delta t, u + \Delta t f)] + O(\Delta t^3)$$

this is the explicit formula of local (truncation) error one order higher than Forward Euler

We have derived expression which indicates that mixing of first derivatives at different times would increase the order of method, let's introduce few parameters to this formula to make it more flexible

$$(1) \quad u(t + \Delta t) = u(t) + \frac{\Delta t}{2} [f(t, u) + f(t + \Delta t, u + \Delta t f)] + O(\Delta t^3)$$

$$(2) \quad u(t + \Delta t) = u(t) + \Delta t [b_1 f(t, u) + b_2 f(t + c\Delta t, u + a\Delta t f)] + O(\Delta t^3)$$

How to find the values of parameters: b_1, b_2, c, a ?

We need to expand $f(t + c\Delta t, u + a\Delta t f)$ to first order, substitute it to (2) and compare it with Taylor series for an exact solution

$$f(t + c\Delta t, u + a\Delta t f) = f + \Delta t [c f_t + f_u f a] + O(\Delta t^2)$$

substitution
keeps
truncation error
unchanged

$$(2a) \quad u(t + \Delta t) = u(t) + \Delta t [(b_1 + b_2) f + \Delta t (b_2 c f_t + b_2 a f f_u)] + O(\Delta t^3)$$

$$\text{exact} \quad u(t + \Delta t) = u(t) + \Delta t f + \frac{\Delta t^2}{2} (f_t + f f_u) + O(\Delta t^3)$$

Comparison of (exact) and (2a) provides us the following conditions

$$b_1 + b_2 = 1, \quad b_2 \cdot c = \frac{1}{2}, \quad b_2 \cdot a = \frac{1}{2}$$

- these conditions must be fulfilled to retain the local error at 3-rd order
- how many are combinations of above parameters? infinite – but usually there are employed only a few

$$(2) \quad u(t + \Delta t) = u(t) + \Delta t [b_1 f(t, u) + b_2 f(t + c\Delta t, u + a\Delta t f)] + O(\Delta t^3)$$

$$\text{conditions: } b_1 + b_2 = 1, \quad b_2 \cdot c = \frac{1}{2}, \quad b_2 \cdot a = \frac{1}{2}$$

Remark: we get 3 equations, but 4 values are needed, hence one parameter is free factor (assumed)
 – it is similar problem to overdetermined systems of equations,
 hence the number of solutions is infinite

Let's find some working RK2 schemes:

- RK2: Heun method (trapezoidal formula)**

$$b_1 = b_2 = \frac{1}{2} \implies c = 1, a = 1$$

$$u_{i+1} = u_i + \frac{\Delta t}{2} [f(t_i, u_i) + f(t_i + \Delta t, u_i + \Delta t f(t_i, u_i))] + O(\Delta t^3)$$

- RK2: midpoint method**

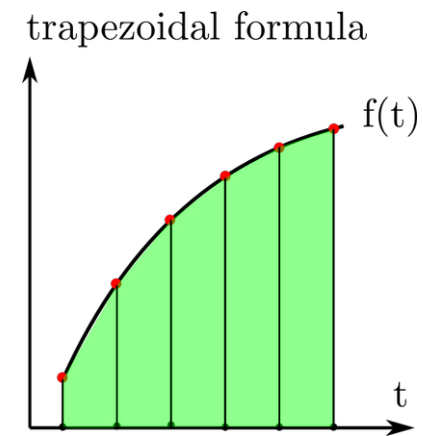
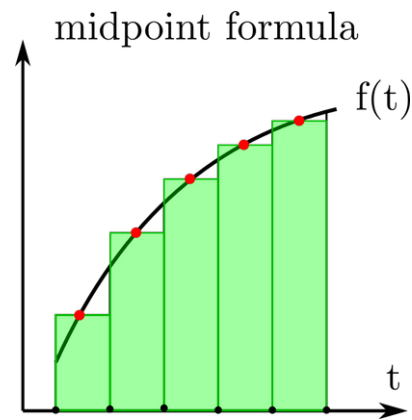
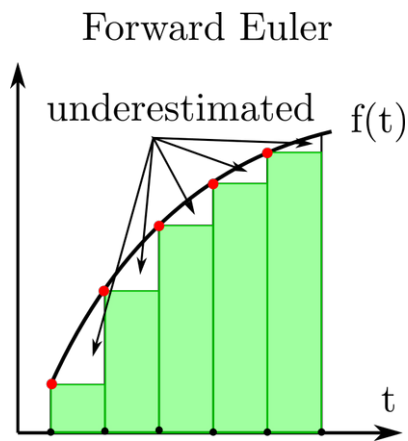
$$b_1 = 0, b_2 = 1 \implies c = \frac{1}{2}, a = \frac{1}{2}$$

$$u_{i+1} = u_i + \Delta t f\left(t_i + \frac{\Delta t}{2}, u_i + \frac{\Delta t}{2} f(t_i, u_i)\right) + O(\Delta t^3)$$

local truncation
error is the same

The difference between 1-st order Forward Euler method and midpoint and trapezoidal 2-nd order methods for the pure time-dependent r.h.s. of ODE

$$\frac{du}{dt} = f(t) \quad \Longrightarrow \quad u_{i+1} = u_0 + \int_{t_0}^{t_n} dt f(t)$$



More common notions for RK2 methods

$$ODE : \quad \frac{du}{dt} = f$$

- Heun method (explicit RK2)

$$k_1 = f(t_i, u_i)$$

$$k_2 = f(t_{i+1}, u_i + \Delta t \cdot k_1)$$

$$u_{i+1} = u_i + \frac{\Delta t}{2} (k_1 + k_2)$$

- midpoint method (explicit RK2)

$$k_1 = f(t_i, u_i)$$

$$k_2 = f\left(t_{i+\frac{1}{2}}, u_i + \frac{\Delta t}{2} k_1\right)$$

$$u_{i+1} = u_i + \Delta t \cdot k_2$$

- it does no matter which one we use to solve ODE, both would give the solutions of similar accuracy
- there is no sense at looking for other RK2 methods since these couldn't give us any other profits
- instead we should look for higher order RK methods – these would really increase the accuracy.

- pseudocode for **RK2 Heun method** used in solving 1-st order ODE $\frac{du}{dt} = f(t, u)$

```

input parameters:
    n, t0, tmax, u0 = u(t0)
initialization:
    Δt ←  $\frac{t_{max} - t_0}{n}$ 
    u ← u0
    t ← t0
iteration:
    for i=1 to n do
        k1 ← f(t, u)
        k2 ← f(t + Δt, u + Δt · k1)
        u ← u +  $\frac{\Delta t}{2} (k_1 + k_2)$ 
        t ← t + Δt
        output: t, u
    end do

```

General formulation of explicit n-th order Runge-Kutta method

- all RK methods are one-time-step methods but during this step the method refers to r.h.s. of ODE several times, these intermediate points are called **stages**
- if the order of RK method is not greater than 4 then number of stages is the same as the (global) order of accuracy ($n=1,2,3,4$ then $s=n$), for higher order methods number of stages increases faster than the order

most economical (?)

order p	1	2	3	4	5	6	7	8	9
stages s	1	2	3	4	6	7	9	11	≥ 12

Definition of **explicit** s-stages p-th order RK method reads

$$\frac{du}{dt} = f(t, u)$$

$$\left\{ \begin{array}{l} k_1 = f(t_i + c_1, u_i) \\ k_2 = f(t_i + c_2, u_i + a_{21} \Delta t k_1) \\ k_3 = f(t_i + c_3, u_i + a_{31} \Delta t k_1 + a_{32} \Delta t k_2) \\ \dots\dots\dots \\ k_s = f(t_i + c_s, u_i + \Delta t \sum_{j=1}^{s-1} a_{sj} k_j) \\ u_{i+1} = u_i + \Delta t \sum_{i=1}^s b_i k_i + O(\Delta t^{p+1}), \quad (s \leq p) \end{array} \right.$$

- to define any RK method we need:

- time stages

$$\vec{c} = [c_1, c_2, c_3, \dots, c_s]$$

- mixing factors for r.h.s. approximations

$$\vec{b} = [b_1, b_2, b_3, \dots, b_s]$$

- contribution factors for better estimations of first derivative

$$A = \begin{bmatrix} 0 & 0 & 0 & \dots & 0 \\ a_{21} & 0 & 0 & \dots & 0 \\ a_{31} & a_{32} & 0 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots \\ a_{s1} & a_{s2} & \dots & a_{s,s-1} & 0 \end{bmatrix}$$

Butcher's tableau

Quite often RK methods are presented in compact tabulated form called the Butcher tableau

$$\begin{array}{c|c} \vec{c} & A \\ \hline & \vec{b}^T \end{array}$$

- 1-st order (Forward Euler)

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

- 2-nd order (midpoint rule)

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

- 2-nd order (Heun method)

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

- 3-th order 31-RK

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

- 3-th order 32-RK

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

- 4-th order classical RK

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

- **4-th order classical RK method** for 1-st order ODE

$$k_1 = f(t_i, u_i)$$

$$k_2 = f\left(t_i + \frac{\Delta t}{2}, u_i + \frac{\Delta t}{2} k_1\right)$$

$$k_3 = f\left(t_i + \frac{\Delta t}{2}, u_i + \frac{\Delta t}{2} k_2\right)$$

$$k_4 = f(t_i + \Delta t, u_i + \Delta t k_3)$$

$$u_{i+1} = u_i + \frac{\Delta t}{6} (k_1 + 2k_2 + 2k_3 + k_4) + O(\Delta t^5)$$

- Butcher's tableau of RK4

0				
$\frac{1}{2}$	$\frac{1}{2}$			
$\frac{1}{2}$	0	$\frac{1}{2}$		
1	0	0	1	
	$\frac{1}{6}$	$\frac{2}{6}$	$\frac{2}{6}$	$\frac{1}{6}$

- this scheme calls r.h.s. 4 times but gives 2 order higher accuracy than RK2, it is very popular, high accuracy, easy to implement on computer

The merits of explicit RK method

- is one-step method, it connects only two subsequent time steps t_i and t_{i+1}
- is self-starting (all one-step methods has this feature)
- is the **general purposes integrator** that can be successfully used in many applications

and the drawbacks

- it is not time-invariant (in some physical simulations would be important)
- it is not symplectic – the **energy drift** is observed for conservative dynamical systems (long-time simulations)

- pseudocode for **system of coupled 1-st order ODE** solved with RK4 method
(simple generalization: single variables are replaced with corresponding vectors)

$$\frac{d\vec{u}}{dt} = \vec{f}(t, \vec{u})$$

```

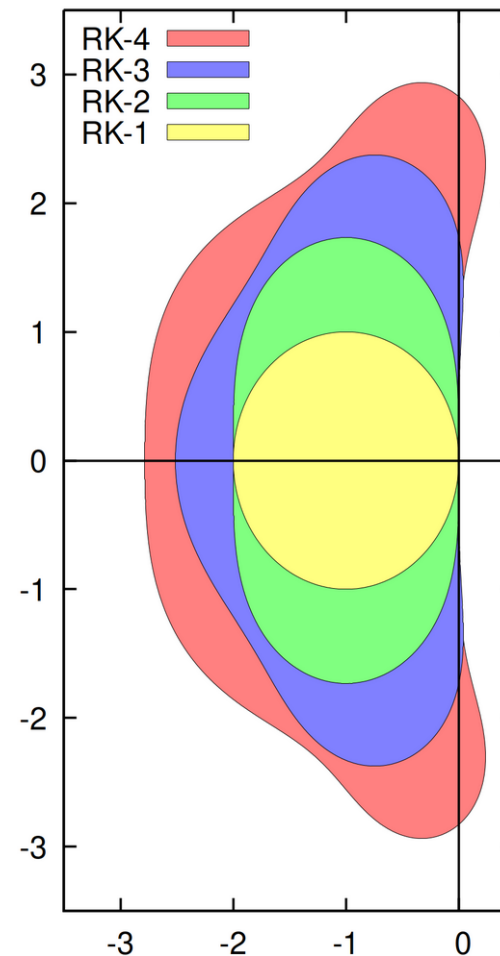
input parameters:
    n, t0, tmax,  $\vec{u}_0 = \vec{u}(t_0)$ 
initialization:
     $\Delta t \leftarrow \frac{t_{max} - t_0}{n}$ 
     $\vec{u} \leftarrow \vec{u}_0$ 
     $t \leftarrow t_0$ 
iteration:
    for i=1 to n do
         $\vec{k}_1 \leftarrow \vec{f}(t, \vec{u})$ 
         $\vec{k}_2 \leftarrow \vec{f}\left(t + \frac{\Delta t}{2}, \vec{u} + \frac{\Delta t}{2} \cdot \vec{k}_1\right)$ 
         $\vec{k}_3 \leftarrow \vec{f}\left(t + \frac{\Delta t}{2}, \vec{u} + \frac{\Delta t}{2} \cdot \vec{k}_2\right)$ 
         $\vec{k}_4 \leftarrow \vec{f}\left(t + \Delta t, \vec{u} + \Delta t \cdot \vec{k}_3\right)$ 
         $\vec{u} \leftarrow \vec{u} + \frac{\Delta t}{6} \left(\vec{k}_1 + 2\vec{k}_2 + 2\vec{k}_3 + \vec{k}_4\right)$ 
         $t \leftarrow t + \Delta t$ 
        output: t,  $\vec{u}$ 
    end do

```

Stability of RK method

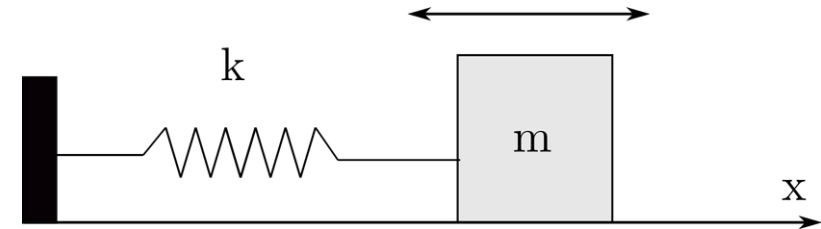
We start to derive the RK2 method by requiring the conformity of difference scheme with first p terms of Taylor series which describes an approximation as p -degree polynomial.
Therefore the stability function of explicit RK method is polynomial

$$R(z) = \underbrace{1 + z}_{p=1} + \underbrace{\frac{z^2}{2}}_{p=2} + \underbrace{\frac{z^3}{6}}_{p=3} + \underbrace{\frac{z^4}{24}}_{p=4} + \dots$$



example: dynamics of 1-dimensional harmonic oscillator

- the spring of constant k is attached to block of mass m , the mass can move horizontally without friction
- this is simple harmonic oscillator problem
- our goal is to find numerically its trajectory $x(t)$, but before we will find its analytical solution



- second Newton's law gives ODE

$$F = -kx = m \frac{d^2x}{dt^2} \quad \Rightarrow \quad \frac{k}{m} = \omega^2 = \text{const} \quad \Rightarrow \quad \frac{d^2x}{dt^2} + \omega^2 x = 0$$

- this harmonic oscillator has analytical solution: position $x(t)$ & velocity $v(t)$

$$x(t) = A \cos(\omega t + \phi_0) \quad v(t) = \frac{dx}{dt} = -A\omega \sin(\omega t + \phi_0)$$

- the system is conservative, that is the total energy E has to be constant during the motion of mass m

$$E = T + U, \quad T = \frac{mv^2}{2}, \quad U = \frac{kx^2}{2}$$

$$E = \frac{kA^2}{2} = \text{const}$$

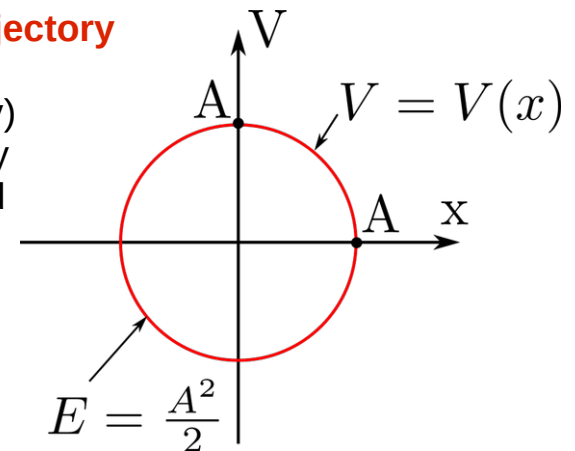
phase-space trajectory

- no time
- only points (x, v) of given energy
- must be closed

$$k = m$$

$$\Downarrow$$

$$\omega = 1$$



The three factors: x , v and E are those we need to properly describe the dynamic of mass m (actually we need only $x(t)$ – the others factors can be derived from it)

Numerical scheme: 2-nd order ODE is transformed into system of two 1-st order ODE

$$\frac{d^2x}{dt^2} + \omega^2 x = 0$$

$$\begin{cases} \frac{dx}{dt} = v \\ \frac{dv}{dt} = -\omega^2 x \end{cases} \implies \begin{cases} u^0 = x \\ u^1 = v \end{cases} \implies \begin{cases} \frac{du^0}{dt} = u^1 \\ \frac{du^1}{dt} = -\omega^2 u^0 \end{cases} \implies \begin{cases} \frac{d\vec{u}}{dt} = \vec{f}(\vec{u}) \\ \vec{u} = [u^0, u^1] \\ \vec{f} = [u^1, -\omega^2 u^0] \end{cases}$$

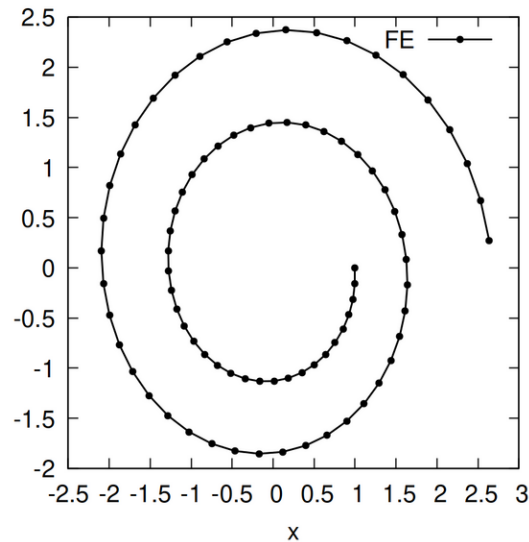
The problem is ready to be solved with Euler and RK methods.

We just need to specify the values of simulation parameters and set the initial condition

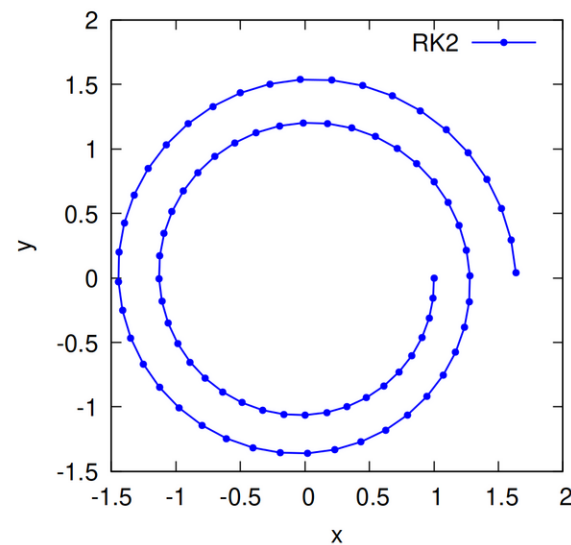
- parameters: $k = m \rightarrow \omega = 1$
- initial condition (displacement without velocity): $\vec{u}_0 = \vec{u}(t_0) = [1, 0]$

- results for $\omega = 1$: $t_{max} = 2T$, $T = \frac{2\pi}{\omega}$, $n = 80$, $\Delta t = \frac{T}{40}$

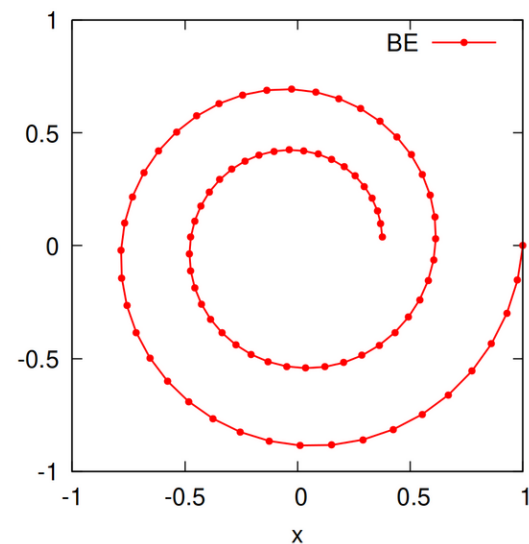
Forward Euler



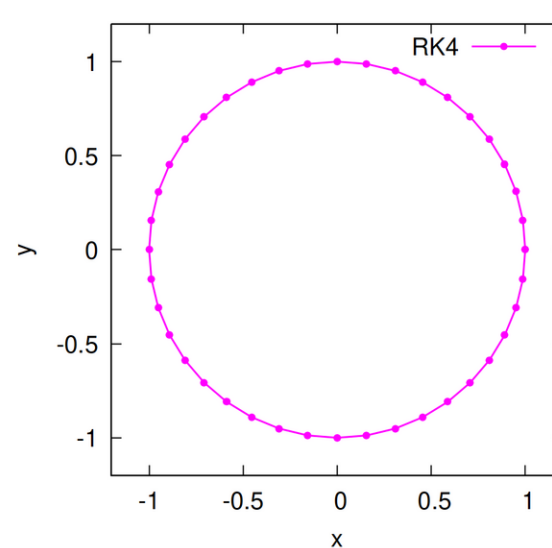
RK2-Heun



Backward Euler



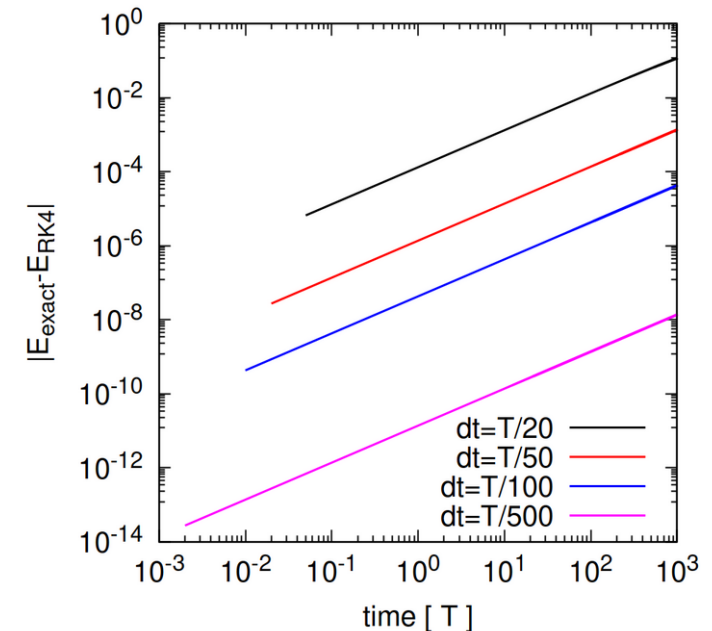
RK4



- non-closed trajectory means the energy is pumped to a system or is gradually taken away
- for given dt only RK4 provides the closed trajectory (at least in this scale)
- the energy error is the smallest for RK4, though for this method it grows with time, we observe

the energy drift
(trajectory is still open)

RK4 - energy error



Simplectic (Hamiltonian) integrators

- instead of usual symbolic representation of components of position and velocity vector we use below the generalized position and momentum variables, for example the cartesian coordinates

$$\begin{aligned}\vec{r} = [x, y, z] &\Rightarrow \vec{q} = [q_1, q_2, q_3] \\ \vec{v} = [v_x, v_y, v_z] &\Rightarrow \vec{p} = [p_1, p_2, p_3] \quad \left(v = \frac{p}{m}\right)\end{aligned}$$

- for any physical system we can always construct the Hamiltonian (energy function) $H(q, p, t)$, we will consider only isolated system $H=H(q, p)$ with separated kinetic $T(p)$ and potential $U(q)$ parts

$$H(\vec{q}, \vec{p}) = T(\vec{p}) + U(\vec{q}) \quad (= \text{total energy} = \text{const})$$

with **equations of motions** (Hamilton-Jacobi equations)

$$\frac{d\vec{q}}{dt} = \frac{\partial H}{\partial \vec{p}} = \nabla_{\vec{p}} H \quad (= \text{velocity}) \quad \frac{d\vec{p}}{dt} = -\frac{\partial H}{\partial \vec{q}} = -\nabla_{\vec{q}} H \quad (= \text{force})$$

- example:** 1-dimensional harmonic oscillator

$$H = \frac{p^2}{2m} + \frac{kq^2}{2} \quad \begin{pmatrix} p = mV \\ q = x \end{pmatrix} \quad \omega^2 = \frac{k}{m}$$

the use of Forward Euler method for q , p and E gives

$$\begin{cases} q_{i+1} = q_i + \Delta t \left. \frac{\partial H}{\partial p} \right|_{t_i} = q_i + \Delta t \frac{p_i}{m} \\ p_{i+1} = p_i - \Delta t \left. \frac{\partial H}{\partial q} \right|_{t_i} = p_i + \Delta t k q_i \end{cases} \Rightarrow \begin{cases} E_i = \frac{p_i^2}{2m} + \frac{kq_i^2}{2} \\ E_{i+1} = \frac{p_{i+1}^2}{2m} + \frac{kq_{i+1}^2}{2} = (1 + \Delta t \omega^2) E_i \end{cases}$$

our goal now is
to remove this
energy drift

Example of symplectic transformation – harmonic oscillator

- the following derivation we will work in the $2n$ -dimensional phase-space so we need the position vector in this space (actually it is a trajectory of the system) and its time-derivative

$$\vec{z}(t) = [q_1(t), q_2(t), \dots, q_N(t), p_1(t), p_2(t), \dots, p_N(t)] = [\vec{q}(t), \vec{p}(t)]$$

$$\frac{d\vec{z}}{dt} = \left[\frac{d\vec{q}}{dt}, \frac{d\vec{p}}{dt} \right] = \left[\frac{\partial H}{\partial \vec{p}}, -\frac{\partial H}{\partial \vec{q}} \right] = \underbrace{\begin{bmatrix} 0 & \mathbf{I}_{N \times N} \\ -\mathbf{I}_{N \times N} & 0 \end{bmatrix}}_{\mathbf{J}_{2N \times 2N}} \begin{bmatrix} \frac{\partial H}{\partial \vec{q}} \\ \frac{\partial H}{\partial \vec{p}} \end{bmatrix} = \mathbf{J} \nabla_{\vec{z}} H \equiv \mathbf{G}(\vec{z}) \cdot \vec{z}$$

- next we define the symplectic form $s(\vec{z}_1, \vec{z}_2)$ which is the small part of phase-space, that is it includes a bundle of neighbouring trajectories

$$s(\vec{z}_1(t), \vec{z}_2(t)) = \vec{z}_1^T(t) \mathbf{J} \vec{z}_2(t) \quad \leftarrow \text{volume in phase-space}$$

$\vec{z}_1(t), \vec{z}_2(t)$ - arbitrary trajectories

The main property of this form is that it remains unaltered under canonical transformations. Canonical transformation do not change the energy and the volume.

In order to convince ourselves of this fact let's consider 1D harmonic problem

For 1D harmonic oscillator we get the solution [q(0),p(0) are initial conditions]

$$H = \frac{p^2}{2m} + \frac{kq^2}{2} \implies \begin{cases} q(t) = q(0) \cos(\omega t) + \frac{p(0)}{m\omega} \sin(\omega t) \\ p(t) = -q(0)m\omega \sin(\omega t) + p(0) \cos(\omega t) \end{cases}$$

we can rewrite solutions of equations of motion to get the vector expression

$$\begin{bmatrix} q(t) \\ p(t) \end{bmatrix} = \underbrace{\begin{bmatrix} \cos(\omega t) & \frac{1}{m\omega} \sin(\omega t) \\ -m\omega \sin(\omega t) & \cos(\omega t) \end{bmatrix}}_{\mathbf{A}(t)} \begin{bmatrix} q(0) \\ p(0) \end{bmatrix}$$

provided **that we know** the global solution, it can be formulated in vector form

$$\vec{z}(t) = \mathbf{A}(t)\vec{z}(0)$$

usually it is not known and we must cope with this problem in different way

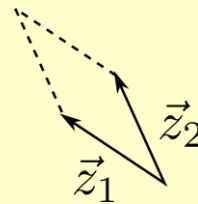
The solution can be generally written as follows

$$\vec{z}(t) = \mathbf{A}(t)\vec{z}(0)$$

- for $t=0$ we have

$$s(\vec{z}_1(0), \vec{z}_2(0)) = \vec{z}_1^T(0) \mathbf{J} \vec{z}_2(0) = [q_1(0), p_1(0)] \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} q_2(0) \\ p_2(0) \end{bmatrix} = q_1(0)p_2(0) - q_2(0)p_1(0)$$

- the volume in phase-space is an area of parallelogram spanned on the vectors $\vec{z}_1$ and $\vec{z}_2$



$$\text{area}(\vec{z}_1, \vec{z}_2) = \begin{vmatrix} q_1 & q_2 & 1 \\ p_1 & p_2 & 1 \\ 0 & 0 & 1 \end{vmatrix} = q_1 p_2 - q_2 p_1$$

- and for some $t > 0$

$$s(\vec{z}_1(t), \vec{z}_2(t)) = \vec{z}_1^T(t) \mathbf{J} \vec{z}_2(t) = \vec{z}_1^T(0) \underbrace{\mathbf{A}^T(t) \mathbf{J} \mathbf{A}(t)}_{=\mathbf{J}} \vec{z}_2(0) = \vec{z}_1^T(0) \mathbf{J} \vec{z}_2(0) = s(\vec{z}_1(0), \vec{z}_2(0)) = \text{const}$$

thus we have proved the volume is preserved

- we get the new condition for canonical transformation

$$\mathbf{A}^T \mathbf{J} \mathbf{A} = \mathbf{J}$$

- property of symplectic transformation

In Hamilton mechanics the time derivative of any dynamical quantity w (e.g. position, momentum, angular momentum) can be formulated by means of **Poisson brackets**

$$w = w(p(t), q(t)) \implies \frac{dw}{dt} = ?$$

$$\frac{dw}{dt} = \frac{\partial w}{\partial q} \frac{dq}{dt} + \frac{\partial w}{\partial p} \frac{dp}{dt} = \underbrace{\frac{\partial w}{\partial q} \frac{dH}{dp} - \frac{\partial w}{\partial p} \frac{dH}{dq}}_{\text{Poisson bracket}} = \{w, H\}$$

which can be used in expression for the time derivative of trajectory in phase-space

$$\frac{d\vec{z}}{dt} = \{\vec{z}, H\} \equiv \mathbf{D}_H(\vec{z}) \cdot \vec{z} \quad - \text{given matrix operator } \mathbf{D}_H \text{ acts on vector } \mathbf{z}$$

this vector ODE equation looks as autonomous problem, so the solution for small time step Δt is

$$\vec{z}(t + \Delta t) = \exp(i\Delta t \mathbf{D}_H) \vec{z}(t)$$

Because the Hamiltonian is separable we can split the Poisson bracket into kinetic and potential energy parts

$$H = T(p) + U(q) \implies \mathbf{D}_H = \mathbf{D}_T + \mathbf{D}_U$$

$$\mathbf{D}_T = \{\cdot, T(\vec{p})\}$$

$$\mathbf{D}_U = \{\cdot, U(\vec{q})\}$$

exploiting this property Nori has proposed approximation of **order p** to the exponential propagator

$$\vec{z}(t + \Delta t) = \underbrace{\prod_{k=1}^p \exp(a_k \Delta t \mathbf{D}_T)}_{\text{second step}} \underbrace{\exp(b_k \Delta t \mathbf{D}_U)}_{\text{first step}} \vec{z}(t) + O(\Delta^{p+1}) \quad \sum_{k=1}^p a_k = \sum_{k=1}^p b_k = 1$$

- pseudocode for single time iteration with symplectic integrator of p-order (here: $p=k_{\max}$)

definitions:

$$\vec{F}(\vec{q}) \equiv -\nabla_{\vec{q}} U(\vec{q})$$

$$\vec{P}(\vec{p}) \equiv \nabla_{\vec{p}} T(\vec{p})$$

initialization: $t_0, t_{\max}, n, \vec{q}_0 = \vec{q}(t_0), \vec{p}_0 = \vec{p}(t_0),$
 $k_{\max} - \text{order}, \Delta t = \frac{t_{\max} - t_0}{n}$

for i=1 to n do

initialization for subiteration:

$$\vec{q}^{(0)} \leftarrow \vec{q}_i$$

$$\vec{p}^{(0)} \leftarrow \vec{p}_i$$

for k=1 to $k_{\max}$ do

$$\vec{q}^{(k)} \leftarrow \vec{q}^{(k-1)} + a_k \vec{P}(\vec{p}^{(k-1)}) \Delta t$$

$$\vec{p}^{(k)} \leftarrow \vec{p}^{(k-1)} + b_k \vec{F}(\vec{q}^{(k)}) \Delta t$$

end do

save new values:

$$\vec{q}_{i+1} \leftarrow \vec{q}^{(p)}$$

$$\vec{p}_{i+1} \leftarrow \vec{p}^{(p)}$$

$$t_{i+1} \leftarrow t_i + \Delta t$$

send output: $t_{i+1}, \vec{q}_{i+1}, \vec{p}_{i+1}$

end do

Parametrization for order p:

- p=1 (Euler-Cromer)

$$a_1 = b_1 = 1$$

- p=2 (Verlet)

$$a_1 = a_2 = \frac{1}{2}$$

$$b_1 = 1, \quad b_2 = 0$$

- p=3 (Ruth)

$$a_1 = \frac{7}{24}, \quad a_2 = \frac{3}{4}, \quad a_3 = -\frac{1}{24}$$

$$b_1 = \frac{2}{3}, \quad b_2 = -\frac{2}{3}, \quad b_3 = 1$$

- p=4 (Neri)

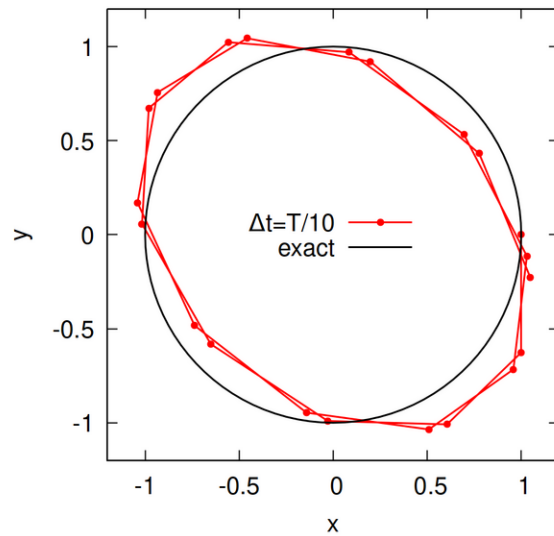
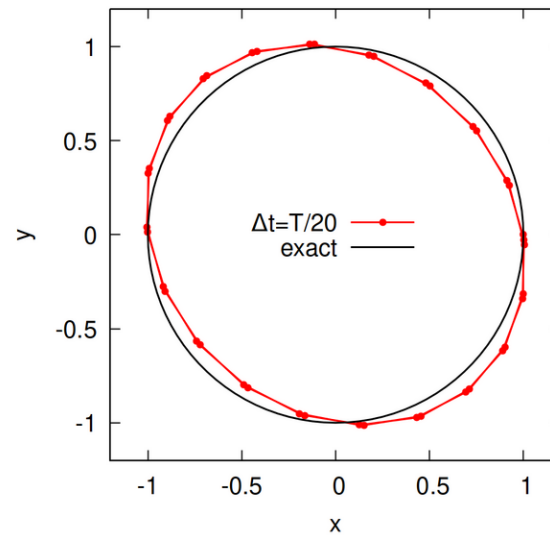
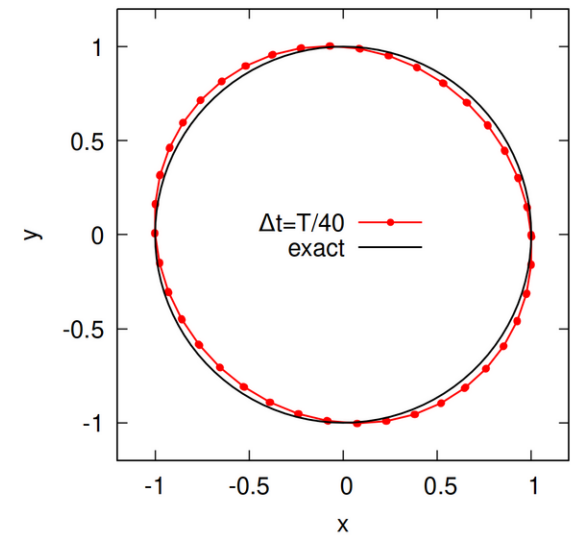
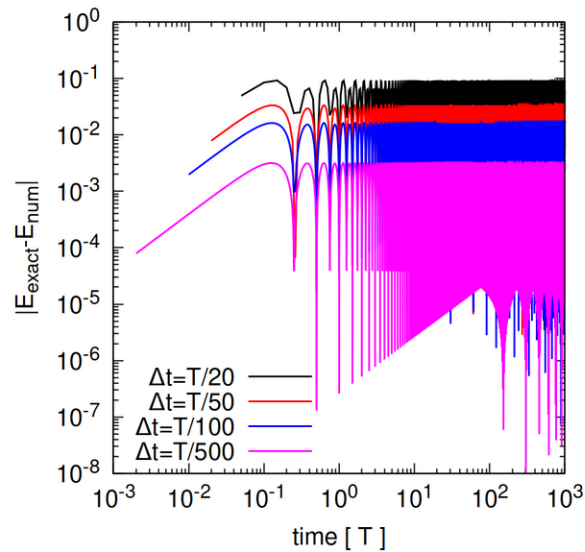
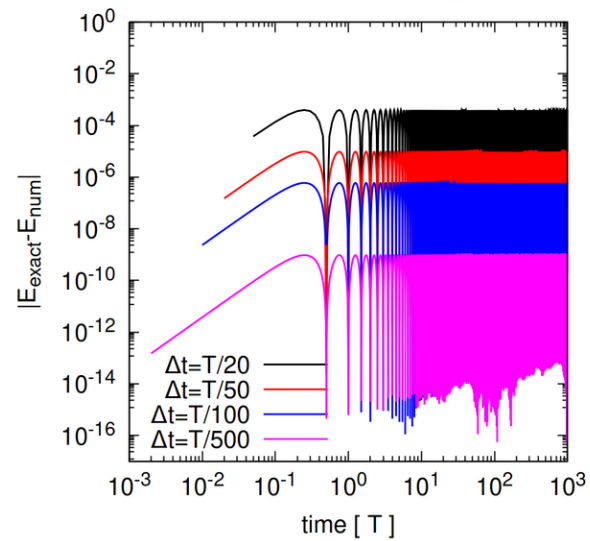
$$a_1 = a_4 = \frac{1}{2(2 - 2^{1/3})}$$

$$a_2 = a_3 = \frac{1 - 2^{1/3}}{2(2 - 2^{1/3})}$$

$$b_1 = b_4 = \frac{1}{(2 - 2^{1/3})}$$

$$b_2 = \frac{-2^{1/3}}{(2 - 2^{1/3})}, \quad b_3 = 0$$

- results obtained for harmonic oscillator with $\omega=1$ T – period of oscillations

SIMPLECTIC: $p=1$, $\Delta t=T/10$, $t_{\max}=2T$ SIMPLECTIC: $p=1$, $\Delta t=T/20$, $t_{\max}=2T$ SIMPLECTIC: $p=1$, $\Delta t=T/40$, $t_{\max}=2T$ SIMPLECTIC: $p=1$ - energy errorSIMPLECTIC: $p=4$ - energy error

RK4 - energy error

