Runge-Kutta Methods

1. Local and Global Errors
   - truncation of Taylor series
   - errors of Euler's method and the modified Euler method
2. Runge-Kutta Methods
   - derivation of the modified Euler method
   - application on the test equation
   - third and fourth order Runge-Kutta methods
3. Applications
   - the pendulum problem
   - the 3-body problem in celestial mechanics

MCS 471 Lecture 30
Numerical Analysis
Jan Verschelde, 1 November 2021

Runge-Kutta Methods

1. Local and Global Errors
   - truncation of Taylor series
   - errors of Euler's method and the modified Euler method
2. Runge-Kutta Methods
   - derivation of the modified Euler method
   - application on the test equation
   - third and fourth order Runge-Kutta methods
3. Applications
   - the pendulum problem
   - the 3-body problem in celestial mechanics

local and global errors

We consider the initial value problem

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

For some step $h > 0$, we set $x_1 = x_0 + h$ and compute

$$y(x_0 + h) = y(x_0) + y'(x_0)h + y''(x_0)\frac{h^2}{2!} + \cdots + O(h^p).$$

The order of the method is p if the approximation coincides with the first p terms of the Taylor series.

The *local error* of a method is the error of one step: $|y_1 - y(x_1)|$.

In the *global error* we take the accumulation of errors into account. After n steps, $x_n = x_0 + nh$, and the global error is $|y_n - y(x_n)|$.

If the local error is $O(h^p)$, then the global error is $O(h^{p-1})$.

Runge-Kutta Methods

1. Local and Global Errors
   - truncation of Taylor series
   - errors of Euler's method and the modified Euler method

2. Runge-Kutta Methods
   - derivation of the modified Euler method
   - application on the test equation
   - third and fourth order Runge-Kutta methods

3. Applications
   - the pendulum problem
   - the 3-body problem in celestial mechanics

Euler's method and the modified Euler method

Consider two examples:

1. Euler's method: $y_{n+1} = y_n + h f(x_n, y_n)$, $n = 0, 1, \ldots$

   Local error: $O(h^2)$, global error: $O(h)$.

2. the modified Euler method:

$$\begin{aligned} \overline{y}_{n+1} &= y_n + h f(x_n, y_n) \\ y_{n+1} &= y_n + \frac{h}{2}\Big(f(x_n, y_n) + f(x_{n+1}, \overline{y}_{n+1}) \Big), \quad n = 0, 1, \ldots \end{aligned}$$

   Local error: $O(h^3)$, global error: $O(h^2)$.

Runge-Kutta Methods

Runge-Kutta methods

Consider a 2-stage Runge-Kutta method:

$$\left\{ \begin{array}{rcl} k_1 & = & f(x_n, y_n) \\ k_2 & = & f(x_n + \alpha h, y_n + \beta k_1) \\ y_{n+1} & = & y_n + a k_1 + b k_2. \end{array} \right.$$

This is a 2-stage method because we have 2 function evaluations.

There are four parameters: α, β, a, b.

The goal is to determine α, β, a, b
so that the order of the method is as high as possible.

a 2-stage Runge-Kutta method

$$\left\{ \begin{array}{rcl} k_1 & = & f(x_n, y_n) \\ k_2 & = & f(x_n + \alpha h, y_n + \beta k_1) \\ y_{n+1} & = & y_n + a k_1 + b k_2 \end{array} \right.$$

is equivalent to

$$y_{n+1} = y_n + a f(x_n, y_n) + b f(x_n + \alpha h, y_n + \beta f(x_n, y_n)).$$

We apply Taylor series in several variables:

$$\begin{array}{l} f(x_n + \alpha h, y_n + \beta f(x_n, y_n)) \\ \quad \approx f(x_n, y_n) + \underbrace{\alpha h}_{\Delta x} f_x(x_n, y_n) + \underbrace{\beta f(x_n, y_n)}_{\Delta y} f_y(x_n, y_n) \end{array}$$

which becomes

$$\begin{array}{rcl} y_{n+1} = y_n & + & a f(x_n, y_n) \\ & + & b \Big(f(x_n, y_n) + \alpha h f_x(x_n, y_n) + \beta f(x_n, y_n) f_y(x_n, y_n) \Big). \end{array}$$

compare the terms with Taylor series of $y(x_n + h)$

$$\begin{aligned} y_{n+1} = y_n & + & a f(x_n, y_n) \\ & + & b\Big(f(x_n, y_n) + \alpha h f_x(x_n, y_n) + \beta f(x_n, y_n) f_y(x_n, y_n) \Big) \end{aligned}$$

$$y_{n+1} = y_n + (a+b) f(x_n, y_n) + \alpha b h f_x(x_n, y_n) + b\beta f(x_n, y_n) f_y(x_n, y_n)$$

Develop $y(x_n + h)$ at x_n with Taylor series:

$$y(x_n + h) = y(x_n) + h y'(x_n) + \frac{h^2}{2} y''(x_n) + O(h^3).$$

We apply this to the differential equation $y' = f$:

$$y_{n+1} = y_n + h f(x_n, y_n) + \frac{h^2}{2} \frac{d}{dx} f(x, y(x)) \Big|_{(x_n, y_n)}.$$

conditions on the parameters

$$y_{n+1} = y_n + hf(x_n, y_n) + \frac{h^2}{2} \frac{d}{dx} f(x, y(x)) \Big|_{(x_n, y_n)}$$

$$\frac{d}{dx} f(x, y(x)) = f_x(x, y(x)) + f_y(x, y(x)) y'(x), \quad y'(x) = f(x, y(x))$$

$$\frac{d}{dx} f(x, y(x)) \Big|_{(x_n, y_n)} = f_x(x_n, y_n) + f_y(x_n, y_n) f(x_n, y_n)$$

$$y_{n+1} = y_n + hf(x_n, y_n) + \frac{h^2}{2} \Big(f_x(x_n, y_n) + f_y(x_n, y_n) f(x_n, y_n) \Big)$$

Compare this to

$$y_{n+1} = y_n + (a + b) f(x_n, y_n) + \alpha b h f_x(x_n, y_n) + b \beta f(x_n, y_n) f_y(x_n, y_n).$$

determination of the parameters

The parameters in the 2-stage Runge-Kutta method

$$y_{n+1} = y_n + (a+b)f(x_n, y_n) + \alpha bh f_x(x_n, y_n) + b\beta f(x_n, y_n) f_y(x_n, y_n)$$

is compared with the Taylor series of the solution at $x_n + h$:

$$y_{n+1} = y_n + hf(x_n, y_n) + \frac{h^2}{2}\left(f_x(x_n, y_n) + f_y(x_n, y_n) f(x_n, y_n) \right).$$

The system on α, β, a, b is

$$\left\{ \begin{array}{rcl} a + b & = & h \\ \alpha bh & = & h^2/2 \\ b\beta & = & h^2/2. \end{array} \right.$$

We can solve this system symbolically with `SymPy`.

solving $a + b = h$, $\alpha bh = h^2/2$, $b\beta = h^2/2$

```
using SymPy
a,b,alpha,beta,h = Sym("a,b,alpha,beta,h")
eq1 = a + b - h
eq2 = alpha*b*h - h^2/2
eq3 = b*beta - h^2/2
sys = [eq1,eq2,eq3]
sol = solve(sys,[a,b,alpha,beta])
println(sol)
```

The output is a 4-tuple of symbolic values for `a`, `b`, `alpha`, and `beta`:

```
[(-h*(-2*beta + h)/(2*beta), h^2/(2*beta), beta/h,
 beta)]
```

Choose $\beta = h$, then $\alpha = 1$, and $a = \dfrac{h}{2} = b$.

the modified Euler method

The solution on the previous slide leads to

$$y_{n+1} = y_n + \frac{h}{2}\left(f(x_n, y_n) + f(x_n + h, y_n + hf(x_n, y_n)) \right),$$

which allows to rewrite the modified Euler method
as a 2-stage Runge-Kutta method:

$$\left\{ \begin{array}{rcl} k_1 & = & f(x_n, y_n) \\ k_2 & = & f(x_n + h, y_n + hk_1) \\ y_{n+1} & = & y_n + \dfrac{h}{2}\left(k_1 + k_2 \right). \end{array} \right.$$

This shows that the modified Euler method has order 3,
which is equivalent to stating that the local error is $O(h^3)$.

the midpoint rule

The modified Euler method is one of the rules of the form

$$y_{n+1} = y_n + (a+b)f(x_n, y_n) + \alpha b h f_x(x_n, y_n) + b\beta f(x_n, y_n) f_y(x_n, y_n)$$

where the parameters satisfy

$$\left\{ \begin{array}{rcl} a+b & = & h \\ \alpha b h & = & h^2/2 \\ b\beta & = & h^2/2. \end{array} \right.$$

Exercise 1:
Set $\alpha = 1/2$ and derive the midpoint method.
Verify the order of the midpoint method.

Runge-Kutta Methods

application on the test equation

$y(x) = \exp(x)$ is the exact solution of the test equation:

$$\frac{dy}{dx} = y, \quad y(0) = 1.$$

Apply the 2-stage Runge-Kutta method

$$\left\{ \begin{array}{rcl} k_1 & = & f(x_n, y_n) \\ k_2 & = & f(x_n + h, y_n + h k_1) \\ y_{n+1} & = & y_n + \dfrac{h}{2} \left(k_1 + k_2 \right) \end{array} \right.$$

with $f = y$:

$$\left\{ \begin{array}{rcl} k_1 & = & y_n \\ k_2 & = & y_n + h k_1 \\ y_{n+1} & = & y_n + \dfrac{h}{2} \left(k_1 + k_2 \right). \end{array} \right.$$

a Julia function

```
"""
    rk2exp(n::Int64)

A 2-stage Runge-Kutta method with n steps
on the interval [0,1] on y' = y, y(0) = 1.
"""
function rk2exp(n::Int64)
    h = 1.0/n
    y0 = 1.0
    y1 = 1.0
    for i=1:n
        x = i*h
        k1 = y0
        k2 = y0 + h*k1
        y1 = y0 + (h/2)*(k1 + k2)
        y0 = y1
    end
    return y1
end
```

on the test equation with $h = 1/10$

```
Running a 2-stage Runge-Kutta method ...
  i   x          k2           2-stage RK        exact         error
  1  0.10  1.100000e+00  1.105000e+00  1.105171e+00  1.71e-04
  2  0.20  1.215500e+00  1.221025e+00  1.221403e+00  3.78e-04
  3  0.30  1.343128e+00  1.349233e+00  1.349859e+00  6.26e-04
  4  0.40  1.484156e+00  1.490902e+00  1.491825e+00  9.23e-04
  5  0.50  1.639992e+00  1.647447e+00  1.648721e+00  1.27e-03
  6  0.60  1.812191e+00  1.820429e+00  1.822119e+00  1.69e-03
  7  0.70  2.002472e+00  2.011574e+00  2.013753e+00  2.18e-03
  8  0.80  2.212731e+00  2.222789e+00  2.225541e+00  2.75e-03
  9  0.90  2.445068e+00  2.456182e+00  2.459603e+00  3.42e-03
 10  1.00  2.701800e+00  2.714081e+00  2.718282e+00  4.20e-03
```

This the same output as the modified Euler method:

- the local error is `1.71e-04`,
- the global error is `4.20e-03`.

application of the midpoint method

Exercise 2:

Apply the midpoint method (see Exercise 1) to the test equation.

Compare the local and global error of the midpoint method to the output of the modified Euler method.

Runge-Kutta Methods

1. Local and Global Errors
   - truncation of Taylor series
   - errors of Euler's method and the modified Euler method

2. Runge-Kutta Methods
   - derivation of the modified Euler method
   - application on the test equation
   - third and fourth order Runge-Kutta methods

3. Applications
   - the pendulum problem
   - the 3-body problem in celestial mechanics

a p-th order Runge-Kutta method

A p-th order Runge-Kutta method proceeds in p stages:

$$\begin{cases} k_1 &= f(x_n, y_n) \\ k_2 &= f(x_n + \alpha_2 h, y_n + \beta_2 k_1) \\ k_3 &= f(x_n + \alpha_3 h, y_n + \beta_3 k_2) \\ &\vdots \\ k_p &= f(x_n + \alpha_p h, y_n + \beta_p k_{p-1}) \\ y_{n+1} &= y_n + a_1 k_1 + a_2 k_2 + a_3 k_3 + \cdots + a_p k_p. \end{cases}$$

If the parameters $\alpha_2, \alpha_3, \ldots, \alpha_p, \beta_2, \beta_3, \ldots, \beta_p$, and $a_1, a_2, a_3, \ldots, a_p$ are determined to agree with the Taylor series of $y(x_n + h)$, then the local error is $O(h^{p+1})$, the global error is $O(h^p)$.

a third order Runge-Kutta method

A third order Runge-Kutta method proceeds in three stages:

$$\left\{ \begin{array}{rcl} k_1 & = & f(x_n, y_n) \\ k_2 & = & f(x_n + h/2, y_n + h\,k_1/2) \\ k_3 & = & f(x_n + 3h/4, y_n + 3\,h\,k_2/4) \\ y_{n+1} & = & y_n + \dfrac{h}{9}\left(2k_1 + 3k_2 + 4k_3\right). \end{array} \right.$$

The local error is $O(h^4)$, the global error is $O(h^3)$.

on the test equation with $h = 1/10$

```
Running a 3-stage Runge-Kutta method ...
  i   x          k3          3-stage RK       exact        error
  1  0.10  1.078750e-01  1.105167e+00  1.105171e+00  4.25e-06
  2  0.20  1.192199e-01  1.221393e+00  1.221403e+00  9.40e-06
  3  0.30  1.317578e-01  1.349843e+00  1.349859e+00  1.56e-05
  4  0.40  1.456143e-01  1.491802e+00  1.491825e+00  2.30e-05
  5  0.50  1.609281e-01  1.648690e+00  1.648721e+00  3.17e-05
  6  0.60  1.778524e-01  1.822077e+00  1.822119e+00  4.21e-05
  7  0.70  1.965565e-01  2.013698e+00  2.013753e+00  5.42e-05
  8  0.80  2.172277e-01  2.225472e+00  2.225541e+00  6.85e-05
  9  0.90  2.400728e-01  2.459518e+00  2.459603e+00  8.52e-05
 10  1.00  2.653205e-01  2.718177e+00  2.718282e+00  1.05e-04
```

For this 3-stage Runge-Kutta method,

- the local error is `4.25e-06`,
- the global error is `1.05e-04`.

a fourth order Runge-Kutta method

A fourth order Runge-Kutta method proceeds in four stages:

$$\left\{ \begin{array}{rcl} k_1 & = & f(x_n, y_n) \\ k_2 & = & f(x_n + h/2, y_n + (h/2)k_1) \\ k_3 & = & f(x_n + h/2, y_n + (h/2)k_2) \\ k_4 & = & f(x_n + h, y_n + h\,k_3) \\ y_{n+1} & = & y_n + \dfrac{h}{6}\left(k_1 + 2k_2 + 2k_3 + k_4\right). \end{array} \right.$$

The local error is $O(h^5)$, the global error is $O(h^4)$.

on the test equation with $h = 1/10$

```
Running a 4-stage Runge-Kutta method ...
  i    x         k4           4-stage RK       exact        error
  1  0.10  1.105250e-01  1.105171e+00  1.105171e+00  8.47e-08
  2  0.20  1.221490e-01  1.221403e+00  1.221403e+00  1.87e-07
  3  0.30  1.349955e-01  1.349858e+00  1.349859e+00  3.11e-07
  4  0.40  1.491931e-01  1.491824e+00  1.491825e+00  4.58e-07
  5  0.50  1.648839e-01  1.648721e+00  1.648721e+00  6.32e-07
  6  0.60  1.822248e-01  1.822118e+00  1.822119e+00  8.38e-07
  7  0.70  2.013896e-01  2.013752e+00  2.013753e+00  1.08e-06
  8  0.80  2.225699e-01  2.225540e+00  2.225541e+00  1.37e-06
  9  0.90  2.459778e-01  2.459601e+00  2.459603e+00  1.70e-06
 10  1.00  2.718474e-01  2.718280e+00  2.718282e+00  2.08e-06
```

For this 4-stage Runge-Kutta method,

- the local error is `8.47e-08`,
- the global error is `2.08e-06`.

summary of the experiments

Running Runge-Kutta methods of order 2, 3, and 4
on the test equation $y' = y$, $y(0) = 1$.

On the interval $[0, 1]$, we do $n = 10$ steps, $h = 1/n = 1/10$.

For a p-stage Runge-Kutta method, we expect

- a local error of $O(h^{p+1})$, and
- a global error of $O(h^p)$.

The actual values for local and global errors are below:

p	local error	global error
2	1.71e-04	4.20e-03
3	4.25e-06	1.05e-04
4	8.47e-08	2.08e-06

The values agree with the expectations.

the test equation with a parameter

For some parameter λ, the initial value problem

$$\frac{dy}{dx} = \lambda y, \quad y(0) = 1$$

has $y(x) = \exp(\lambda x)$ as the exact solution.

Exercise 3:

1. Take $\lambda = 0.1$ and consider the interval $[0, 1]$.
   For $h = 0.1$, run the 2-stage Runge-Kutta method.
   Compare the local and global error with the errors on the test equation without parameter (or for $\lambda = 1$).

2. Take $\lambda = 0.01$ and consider the interval $[0, 1]$.
   For $h = 0.1$, run the 4-stage Runge-Kutta method.
   Compare the results with your findings of the previous part.

Runge-Kutta Methods

1. Local and Global Errors
   - truncation of Taylor series
   - errors of Euler's method and the modified Euler method
2. Runge-Kutta Methods
   - derivation of the modified Euler method
   - application on the test equation
   - third and fourth order Runge-Kutta methods
3. Applications
   - the pendulum problem
   - the 3-body problem in celestial mechanics

the pendulum problem

The initial value problem to model a pendulum is

$$\begin{array}{rcl} y_1' & = & y_2, \\ y_2' & = & -\dfrac{g}{\ell}\sin(y_1), \quad y_1 = \pi/4, \quad y_2 = 0. \end{array}$$

We apply a vector version of a 2-stage Runge-Kutta method:

$$\left\{ \begin{array}{rcl} \mathbf{k}_1 & = & \mathbf{f}(x_n, \mathbf{y}_n) \\ \mathbf{k}_2 & = & \mathbf{f}(x_n + h, \mathbf{y}_n + h\mathbf{k}_1) \\ \mathbf{y}_{n+1} & = & \mathbf{y}_n + \dfrac{h}{2}\left(\mathbf{k}_1 + \mathbf{k}_2\right). \end{array} \right.$$

where

$$\mathbf{y}_n = \left[\begin{array}{c} y_{n,1} \\ y_{n,2} \end{array} \right], \ \mathbf{k}_1 = \left[\begin{array}{c} k_{1,1} \\ k_{1,2} \end{array} \right], \ \mathbf{k}_2 = \left[\begin{array}{c} k_{2,1} \\ k_{2,2} \end{array} \right], \ \mathbf{f} = \left[\begin{array}{c} y_2 \\ (-g/\ell)\sin(y_1) \end{array} \right].$$

a 2-stage Runge-Kutta method for the pendulum

$$\mathbf{y}_n = \left[\begin{array}{c} y_{n,1} \\ y_{n,2} \end{array} \right], \ \mathbf{k}_1 = \left[\begin{array}{c} k_{1,1} \\ k_{1,2} \end{array} \right], \ \mathbf{k}_2 = \left[\begin{array}{c} k_{2,1} \\ k_{2,2} \end{array} \right], \ \mathbf{f} = \left[\begin{array}{c} y_2 \\ (-g/\ell)\sin(y_1) \end{array} \right]$$

$$\left\{ \begin{array}{rcl} \mathbf{k}_1 & = & \mathbf{f}(x_n, \mathbf{y}_n) \\ \mathbf{k}_2 & = & \mathbf{f}(x_n + h, \mathbf{y}_n + h\mathbf{k}_1) \\ \mathbf{y}_{n+1} & = & \mathbf{y}_n + \frac{h}{2}\left(\mathbf{k}_1 + \mathbf{k}_2\right) \end{array} \right.$$

$$\left[\begin{array}{c} k_{1,1} \\ k_{1,2} \end{array} \right] = \left[\begin{array}{c} y_2 \\ (-g/\ell)\sin(y_1) \end{array} \right], \ \left[\begin{array}{c} k_{2,1} \\ k_{2,2} \end{array} \right] = \left[\begin{array}{c} y_2 + h k_{1,\mathbf{2}} \\ (-g/\ell)\sin(y_1 + h k_{1,\mathbf{1}}) \end{array} \right],$$

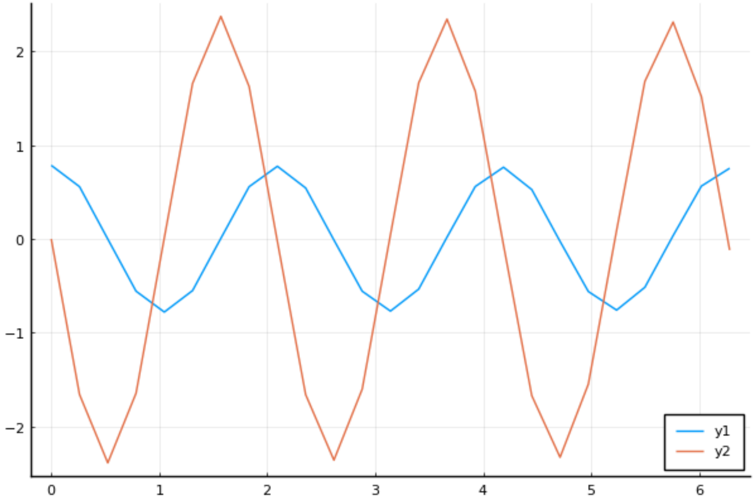
$$\left[\begin{array}{c} y_{n+1,1} \\ y_{n+1,2} \end{array} \right] = \left[\begin{array}{c} y_{n,1} \\ y_{n,2} \end{array} \right] + \frac{h}{2}\left(\left[\begin{array}{c} k_{1,1} \\ k_{1,2} \end{array} \right] + \left[\begin{array}{c} k_{2,1} \\ k_{2,2} \end{array} \right] \right)$$

running 24 steps

```
$ julia rkpend.jl
Running the modified Euler method ...
    1    0.00    7.853982e-01    0.000000e+00   0.00e+00
   25    6.28    1.550520e+00    2.841312e+00   7.65e-01
Running a 2-stage Runge-Kutta method ...
    1    0.00    7.853982e-01    0.000000e+00   0.00e+00
   25    6.28    1.550520e+00    2.841312e+00   7.65e-01
Running a 3-stage Runge-Kutta method ...
    1    0.00    7.853982e-01    0.000000e+00   0.00e+00
   25    6.28    5.106470e-01   -7.919150e-01   2.75e-01
Running a 4-stage Runge-Kutta method ...
    1    0.00    7.853982e-01    0.000000e+00   0.00e+00
   25    6.28    7.543960e-01   -1.190943e-01   3.10e-02
```

The columns are respectively the step number n,
the value for t_n, $y_{n,1}$ (position), $y_{n,2}$ (velocity), and the error.
As expected, the errors decrease as the order increases.

24 steps with a 4-stage Runge-Kutta method



apply Runge-Kutta methods

Exercise 4:

Apply Runge-Kutta methods of order three and four to

$$\left\{ \begin{array}{rcl} \frac{dx}{dt} & = & -y + \cos(t)\sin(t) \\ \frac{dy}{dt} & = & x + \sin^2(t) \end{array} \right. \quad x(0) = 0, y(0) = 0,$$

for $t \in [0, 2\pi]$. Its exact solution is the cardioid.

1. For both the order three and four Runge-Kutta methods, set h to be small enough so the plot of the computed points agrees with the plot of the exact solution.
2. Compare the accuracy of both runs. What is the largest h you can use with the fourth order method and achieve the same accuracy as with the third order?

another plane curve

Exercise 5:

Apply Runge-Kutta methods of order three and four to

$$\left\{ \begin{array}{rcl} \frac{dx}{dt} & = & -y + 3\cos(3t)\cos(t) \\ \frac{dy}{dt} & = & x + 3\cos(3t)\sin(t) \end{array} \right. \quad x(0) = 0, y(0) = 0,$$

for $t \in [0, 2\pi]$. See the previous lecture for its exact solution.

1. For both the order three and four Runge-Kutta methods, set h to be small enough so the plot of the computed points agrees with the plot of the exact solution.
2. Compare the accuracy of both runs.
   What is the largest h you can use with the fourth order method and achieve the same accuracy as with the third order?

Runge-Kutta Methods

1. Local and Global Errors
   - truncation of Taylor series
   - errors of Euler's method and the modified Euler method

2. Runge-Kutta Methods
   - derivation of the modified Euler method
   - application on the test equation
   - third and fourth order Runge-Kutta methods

3. Applications
   - the pendulum problem
   - the 3-body problem in celestial mechanics

the 3-body problem

We consider three bodies with respective masses m_1, m_2, m_3 in the plane with positions $(x_1(t), y_1(t))$, $(x_2(t), y_2(t))$, $(x_3(t), y_3(t))$ evolving over time t, governed by a system of second order differential equations, shown below for the movement of the first body:

$$\begin{aligned}\frac{d^2 x_1(t)}{dt^2} = & -\frac{m_2(x_1(t) - x_2(t))}{\left((x_1(t) - x_2(t))^2 + (y_1(t) - y_2(t))^2\right)^{3/2}} \\ & -\frac{m_3(x_1(t) - x_3(t))}{\left((x_1(t) - x_3(t))^2 + (y_1(t) - y_3(t))^2\right)^{3/2}}\end{aligned}$$

The equation for $y_1(t)$ is similar (replace x in numerator by y).

With four additional equations for the positions of the second and third body, our model consists of six second order equations.

a system of 12 first order equations

To turn this into a system of first order differential equations
we introduce new variables u_i, v_i for the velocities of x_i, y_i so we have

$$
\begin{aligned}
\frac{dx_1(t)}{dt} &= u_1(t) \\
\frac{du_1(t)}{dt^2} &= -\frac{m_2(x_1(t) - x_2(t))}{\left((x_1(t) - x_2(t))^2 + (y_1(t) - y_2(t))^2\right)^{3/2}} \\
& \quad -\frac{m_3(x_1(t) - x_3(t))}{\left((x_1(t) - x_3(t))^2 + (y_1(t) - y_3(t))^2\right)^{3/2}}
\end{aligned}
$$

So we obtain 12 first order differential equations.

See the posted Julia program and Jupyter notebook.

a figure eight