Numerical solution of ordinary differential equations

Files for the present tutorial can be found using following links:

- Header file rk4.h
- Source file rk4.cpp

1. Introduction

The goal of the present laboratory is to introduce the basic methods used for solution (integration) of the first order ordinary differential equations (ODE's). Differential equations are used for mathematical modelling of almost all physical phenomena we accounter in the real world. Most of the equations derived for description of the given phenomena are to complex to be solved analytically. In particular, the most difficult to solve are equations including non-linear terms. Interestingly, almost all physical phenomena in reality (and their models too) are non-linear hence the world is non-linear (complex) place ...

As engineer needs solutions and hence anlytical solution of the problems are usually not posssible, we often use numerical methods to obtain approximate solutions of complex problems. The basic example here is the Euler's method that is the simplest, first-order accurate technique for solution of ODE's. The second approach to the discussed problem will be carried out using 4-th order accurate Runge-Kutta method. It is an example of multi-step method with high order of accuracy and relatively higher stability in comparison to the Euler's method.

2. Explicit Euler's method

Euler's method will be used to solve initial value problem in the form:

$$\frac{dy}{dt} = f(t, y)$$

$$y(t_0) = y_0$$

Using following iterative scheme:

$$t_{i+1} = t_i + h$$

$$y_{i+1} = y_i + h \cdot f(t_i, y_i)$$

where h – is the size of integration step, y_{i+1} – solution at $i+1$ time level, y_i – solution on the previous $i-th$ time level and f - function defining the right hand side of the given ODE.

3. Exercises

Consider the following initial value problem:

$$\begin{cases} \frac{dy}{dt} & = & \lambda \cdot y(t) \\ y(t_0) & = & y_0 \end{cases}$$

The exact solution to this problem has a form:

$$y(t) = y_0 e^{\lambda(t-t_0)}$$

1. Write a program solving given initial value problem using Euler's method.
2. Write a program solving given initial value problem using RK4.
3. In both cases, display on the screen following values of t_i, y_i and the error defined as $\varepsilon_i = \frac{|y_i - y_{i,\text{analytical}}|}{|y_{i,\text{analytical}}|}$.
4. Modify the program in a way it makes computations using both methods for a given time step size h or for the given number of time steps $2^0, 2^1, \ldots, 2^6$. $T = n \cdot h$ is the total intergration time (set by the user).
   Print to the file: number of time steps, time step h, error of the Euler and RK4 methods *only* for the last time step.
5. Prepare diagram showing errors of both methods dependent of the time step size h and estimate the order of their convergence.