Całkowanie numeryczne równań różniczkowych zwyczajnych

Pliki do wykorzystania w poniższym ćwiczeniu można pobrać za pomocą poniższych linków:

- Plik nagłówkowy rk4.h
- Plik źródłowy rk4.cpp

1. Wstęp

Celem dzisiejszych zajęć jest zapoznanie się z podstawowymi metodami całkowania numerycznego równań różniczkowych zwyczajnych pierwszego rzędu. Część procesów fizycznych, które obserwujemy w otaczającym nas świecie może być modelowana za pomocą właśnie równań różniczkowych. Większości z nich nie da się rozwiązać w sposób analityczny (tj. podać rozwiązania w postaci jawnej). W szczególności, największe problemy sprawiają równania zawierające człony nieliniowe. Świat, w którym żyjemy, jest silnie nieliniowy i większość problemów, które przyjdzie nam rozwiązywać, nie będzie posiadać rozwiązania w formie analitycznej.

Jako pierwszą poznamy metodę pierwszego rzędu zwaną (od nazwiska twórcy) jawną metodą **Eulera**. Jest ona jednocześnie najprostszą i bardzo niestabilną metodą. Drugą metodą, którą zastosujemy na dzisiejszych zajęciach, będzie metoda **Rungego-Kutty 4-go rzędu**. Jest to metoda jawna, która charakteryzuje się stosunkowo wysokim rzędem, łatwością implementacji oraz relatywnie wysoką stabilnością.

Obie metody wykorzystamy do rozwiązania zagadnienia początkowego w postaci:

$$\begin{cases} \frac{dy}{dt} = f(t, y) \\ y(t_0) = y_0 \end{cases}$$

2. Metoda Eulera

W celu rozwiązania zastosujmy następujący schemat iteracyjny:

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

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

Gdzie:

- h – krok całkowania,
- y_{i+1} – rozwiązanie,
- y_i – rozwiązanie w kroku poprzednim,
- f - funkcja obliczająca prawą stronę równania różniczkowego.

3. Metoda Rungego-Kutty 4-ego rzędu

Schemat iteracyjny ma postać:

$$y_{i+1} = y_i + \frac{h}{6}(K_1 + 2K_2 + 2K_3 + K_4)$$

gdzie:

$$K_1 = f(x_i, y_i)$$

$$K_2 = f\left(x_i + \frac{1}{2}h, y_i + \frac{1}{2}hK_1\right)$$

$$K_3 = f\left(x_i + \frac{1}{2}h, y_i + \frac{1}{2}hK_2\right)$$

$$K_4 = f(x_i + h, y_i + hK_3)$$

Metoda ta **jest już** zaimplementowana w pliku `rk4.cpp`. Nagłówek funkcji, która wykonuje jeden krok całkowania równania różniczkowego ma postać:

```
double rk4(double x0, double y0, double h, double (*fun)(double, double))
```

gdzie:

- `x0` – wartość początkowa zmiennej niezależnej,
- `y0` – wartość początkowa zmiennej zależnej,
- `h` – krok całkowania,
- `fun` – adres funkcji obliczającej prawe strony równania.

4. Ćwiczenia

Dane jest zagadnienie początkowe w postaci:

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

Jego rozwiązanie dokładne to:

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

1. Napisz program, który rozwiąże dane zagadnienie początkowe z wykorzystaniem schematu Eulera.
2. Wykorzystaj schemat RK4 zaimplementowany w pliku `rk4.cpp` i ponownie rozwiąż zagadnienie.
3. Dla obu przypadków wyświetl na monitorze kolejne wartości t_i, y_i oraz względne wartości błędów: $\varepsilon_i = \frac{|y_i - y_i^{\text{analityczne}}|}{|y_i^{\text{analityczne}}|}$.
4. Zmodyfikuj program tak aby wykonywał obliczenia jedną i druga metodą dla zadanego t_k i liczby kroków: $N = 2^0,\ 2^1,\ \ldots,\ 2^6$.
   Wydrukuj do pliku: liczbę kroków N, długość kroku h, błąd metody Eulera i błąd metody RK4 dla ostatniego kroku czasowego.
5. Sporządź wykresy błędów obu metod w funkcji kroku h i oszacuj ich rzędy zbieżności.