

# Sprawozdanie

## Metody Numeryczne 2, laboratorium 6

Grzegorz Rozdzialik (D4, grupa lab. 2)

7 stycznia 2017

### 1 Zadanie

Temat **6**, zadanie **11**:

Metoda Rungego-Kutty rzędu 4-go (wzór "3/8") dla układu dwóch równań.

Niech  $x \in [a, b] \subset \mathbb{R}$ . Dane zagadnienie początkowe do rozwiązania:

$$\begin{cases} y_1' &= f_1(x, y_1, y_2) \\ y_2' &= f_2(x, y_1, y_2) \\ y_1(a) &= y_{1,a} \\ y_2(a) &= y_{2,a} \end{cases} \quad (1)$$

Niech  $n \in \mathbb{N}$  będzie liczbą oznaczającą ilość podprzedziałów, na jakie należy podzielić odcinek  $[a, b]$  tak, że:

$$\begin{aligned} h &= \frac{b-a}{n} \\ x_0 &= a \\ x_i &= x_{i-1} + h \quad \text{dla } i = 1, 2, \dots, n \end{aligned}$$

## 2 Opis metody

Metoda Rungego-Kutty rzędu 4 dla pojedynczego równania różniczkowego określa się wzorem:

$$y_{n+1} = y_n + \sum_{i=1}^4 w_i K_i \quad (2)$$

gdzie:

$$K_1 = hf(x_n, y_n)$$

$$K_i = hf(x_n + a_i h, y_n + \sum_{j=1}^{i-1} b_{i,j} K_j), \quad i > 1$$

$w_i, a_i, b_{i,j}$  - stałe.

Dla metody "trzech ósmych" te stałe wynoszą:

$$w = \begin{bmatrix} \frac{1}{8} & \frac{3}{8} & \frac{3}{8} & \frac{1}{8} \end{bmatrix}^T$$
$$a = \begin{bmatrix} 0 & \frac{1}{3} & \frac{2}{3} & 1 \end{bmatrix}^T$$
$$b = \begin{bmatrix} 0 & 0 & 0 \\ 1/3 & 0 & 0 \\ -1/3 & 1 & 0 \\ 1 & -1 & 1 \end{bmatrix}$$

Licząc rozwiązanie w punkcie  $x_i, i = 1, 2, \dots, n$  należy oddzielnie obliczać wektor  $K$  dla  $y_1$  oraz  $y_2$ .

## 3 Implementacja metody

Implementacja metody podzielona jest na następujące kroki:

1. `[x, y1, y2] = solveDifferentialSystem(f1, f2, a, b, n, y1a, y2a)` - podział odcinka  $[a, b]$  na  $n$  podprzedziałów, wykorzystanie zagadnienia początkowego do uzupełnienia  $y_{1,a}, y_{2,a}$ , iteracyjne obliczenie kolejnych rozwiązań funkcją `calculateSolutionValue`

2. `[y1n, y2n] = calculateSolutionValue(f1, f2, y1, y2, x, n)` - obliczenie rozwiązań  $y_{1,n}$  oraz  $y_{2,n}$  wykorzystując metodę Rungego Kutty rzędu 4 (wzór (2))
3. `plotResults(x, y1, y2, y1Exact, y2Exact, joinPlots)` - naniesienie wartości rozwiązania obliczonych metodą (2) oraz wartości dokładnych na wykres. Dodatkowo utworzenie wykresu z modułem błędów dla obu metod.

## 4 Poprawność metody

Z rozdziału dotyczącego *Metod Rungego-Kutty* w książce Z. Fortuna, B. Macukow, J. Wąsowski *Metody numeryczne* wiadomo, że wzór "trzech ósmych" jest absolutnie stabilny dla dostatecznie małych  $h$ , zatem zwiększając parametr  $n$ , a co za tym idzie zmniejszając długość każdego z podprzedziałów (czyli zmniejszając  $h$ ) otrzymujemy absolutną stabilność tej metody.

Można to łatwo zauważyć zwiększając parametr  $n$  w skrypcie testowym (opisanym w sekcji 7).

## 5 Przykłady

W przykładach znajdują się fragmenty konfiguracji skryptu do testowania metody (opisany w sekcji 7).

Przykładowo, zagadnieniu początkowemu:

$$\begin{cases} y_1' &= y_2 \\ y_2' &= 3e^{\frac{x}{2}} - y_2 \\ y_1(0) &= 4 \\ y_2(0) &= 3 \end{cases}$$

odpowiada następująca konfiguracja:  
`f1 = @(x, y1, y2)(y2);`

```

f2 = @(x, y1, y2)(3 * exp(x / 2) - y2);
a = 0;
b = 10;
y1a = 4;
y2a = 3;

```

**Przykład 1** Zagadnienie początkowe utworzone z przekształcenia równania różniczkowego 2-go rzędu na układ równań różniczkowych 1-go rzędu.

$$y'' + y' = 3e^{\frac{t}{2}}$$

$$\begin{cases} y_1' &= y_2 \\ y_2' &= 3e^{\frac{t}{2}} - y_2 \\ y_1(0) &= 4 \\ y_2(0) &= 3 \end{cases}$$

Znane rozwiązanie:

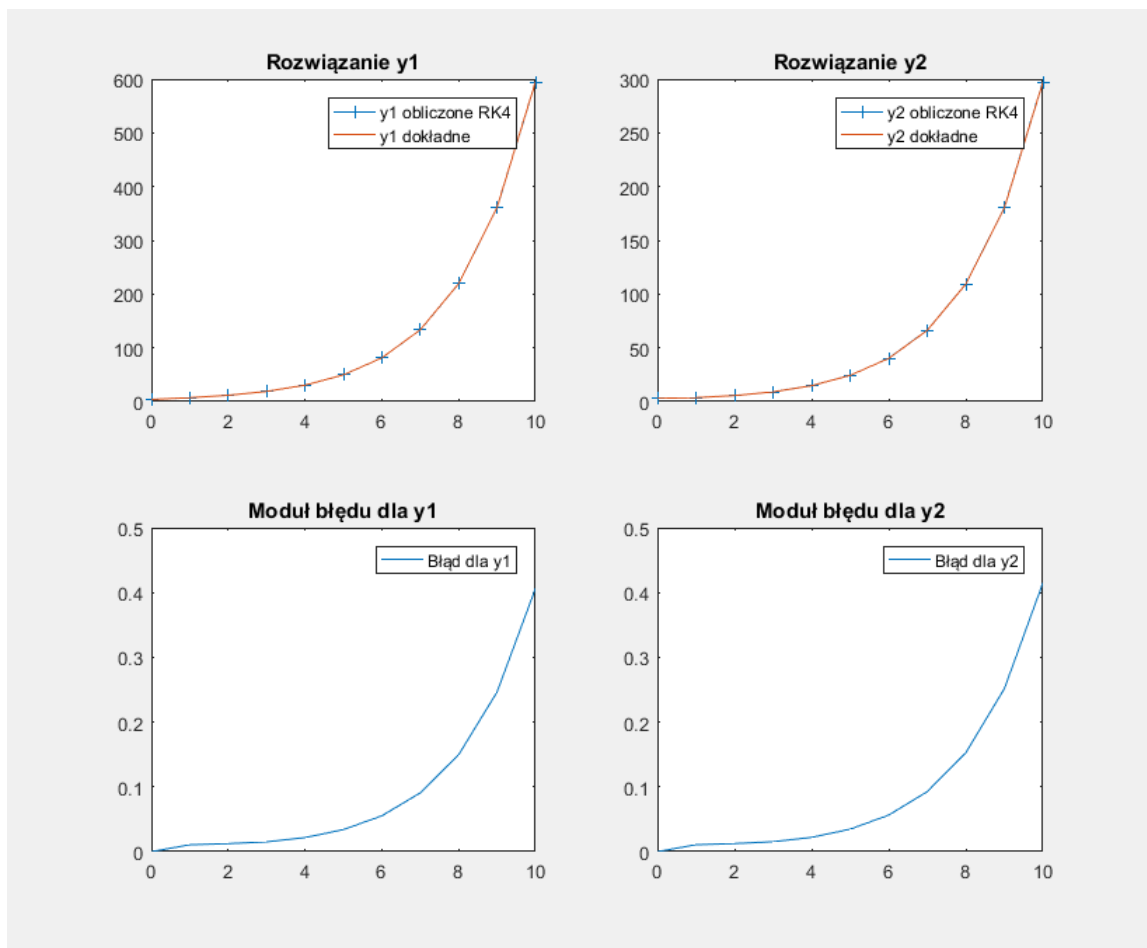
$$\begin{cases} y_1(x) &= -e^{-x} + 4e^{\frac{x}{2}} + 1 \\ y_2(x) &= e^{-x} + 2e^{\frac{x}{2}} \end{cases}$$

Konfiguracja skryptu:

```

f1 = @(x, y1, y2)(y2);
f2 = @(x, y1, y2)(3 * exp(x / 2) - y2);
a = 0;
b = 10;
y1a = 4;
y2a = 3;
y1Solution = @(x)(-exp(-x) + 4 * exp(x / 2) + 1);
y2Solution = @(x)(exp(-x) + 2 * exp(x / 2));
n = 10

```



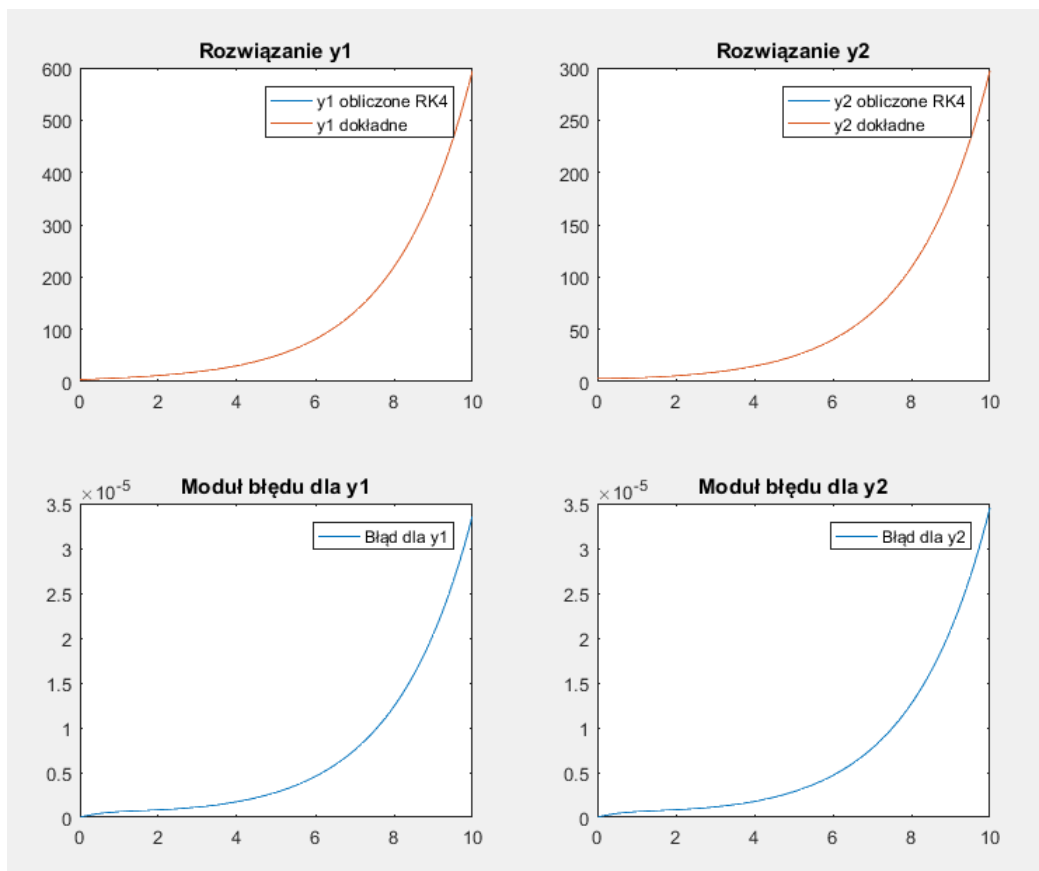
Czas rozwiązywania zagadnienia początkowego: 0.000003 ms

Maksymalny moduł błędu między rozwiązaniem dokładnym a obliczonym metodą:

- dla y1: 0.4060596
- dla y2: 0.4145344

Jak widać obliczenia zajmują bardzo mało czasu i są dość dokładne już dla małych wartości  $n$ .

**Przykład 2** To samo zagadnienie początkowe i konfiguracja co w poprzednim przykładzie, jednakże 10-krotnie zwiększona liczba przedziałów:  $n = 100$



Czas rozwiązywania zagadnienia początkowego: 0.000006 ms  
Maksymalny moduł błędu między rozwiązaniem dokładnym a obliczonym metodą:

- dla  $y_1$ :  $3.362839 \cdot 10^{-5}$
- dla  $y_2$ :  $3.448142 \cdot 10^{-5}$

Maksymalny błąd zmalał 10000-krotnie, a czas obliczeń wzrósł jedynie 2-krotnie.

**Przykład 3** To samo zagadnienie początkowe co w poprzednim przykładzie.

10-krotnie zwiększona liczba przedziałów:  $n = 1000$

Wykresy wyglądają bardzo podobnie, zmienia zmienia się jedynie skala na wykresie błędów, więc nie zostały tu umieszczone, ponieważ nic nie

wnoszą do przykładu.

Czas rozwiązywania zagadnienia początkowego: 0.000041 ms

Maksymalny moduł błędu między rozwiązaniem dokładnym a obliczonym metodą:

- dla  $y_1$ :  $3.274067 * 10^{-9}$
- dla  $y_2$ :  $3.359560 * 10^{-9}$

Znowu błąd zmalał 10000-krotnie. Czas obliczeń tym razem wzrósł około 7 razy.

**Przykład 4** Inny układ równań różniczkowych.

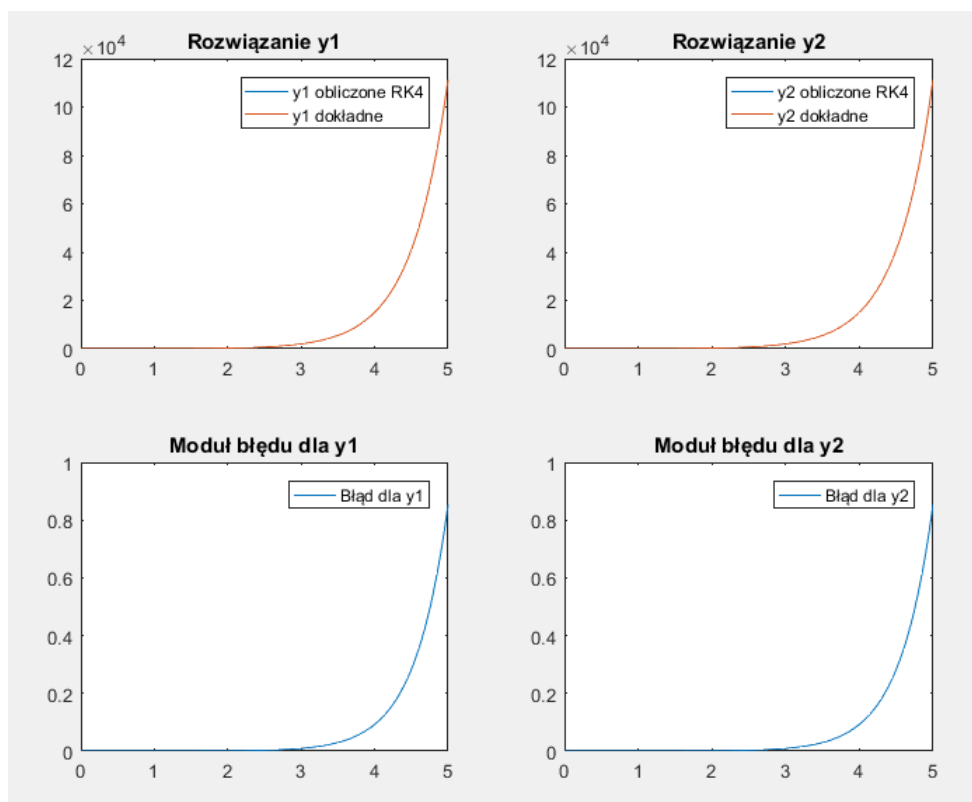
$$\begin{cases} y_1' &= y_1 + y_2 + x \\ y_2' &= 3y_1 - y_2 + 2x \\ y_1(0) &= 5 \\ y_2(0) &= 4 \end{cases}$$

Znane rozwiązanie:

$$\begin{cases} y_1(x) &= \frac{1}{16}(-12x + 3e^{-2x} + 81e^{2x} - 4) \\ y_2(x) &= \frac{1}{16}e^{-2x}(-4e^{2x}(x + 2) + 81e^{4x} - 9) \end{cases}$$

Konfiguracja skryptu:

```
f1 = @(x, y1, y2)(y1 + y2 + x);
f2 = @(x, y1, y2)(3 * y1 - y2 + 2 * x);
a = 0;
b = 5;
y1a = 5;
y2a = 4;
y1Solution = @(x)((-12 .* x + 3 * exp(-2 .* x) + 81 * exp(2
* x) - 4) / 16);
y2Solution = @(x)((exp(-2 * x) / 16 .* (-4 * exp(2 * x) .* (x
+ 2) + 81 * exp(4 * x) - 9)));
n = 100
```



Czas rozwiązywania zagadnienia początkowego: 0.000004 ms

Maksymalny moduł błędu między rozwiązaniem dokładnym a obliczonym metodą:

- dla y1: 0.8550255
- dla y2: 0.8550255

Przy krótkim czasie obliczeń udało się osiągnąć dość dobrą dokładność, ponieważ przy rozwiązaniach o wartościach rzędu  $10^4$  maksymalny moduł błędu wyniósł mniej niż 1.

## 6 Wnioski

1. Na podstawie porównań maksymalnego modułu błędu można zauważyć, że metoda (2) jest faktycznie rzędu 4, ponieważ przy 10-krotnym



zwiększeniu liczby przedziałów błąd zmalał 10000-krotnie (**Przykład 1** oraz **Przykład 2**).

2. Metoda ma złożoność  $O(n)$ . Można to zauważyć zwiększając parametr  $n$ . Ponadto, w metodzie występuje jedynie jedna pętla zależna od ilości podprzedziałów, w której jest stała liczba obliczeń (w każdej iteracji).
3. Metoda działa bardzo szybko nawet dla dużych  $n$  ( $n = 1000$  w przykładzie **Przykład 3**).
4. Zaletami tej metody są prosta implementacja, szybkość działania oraz dobra dokładność.

## 7 Skrypt do testowania metody

Do testowania metody został utworzony skrypt znajdujący się w pliku `textScript.m`.

Pozwala on na wpisanie własnego zagadnienia początkowego, rozwiązania tego zagadnienia (w celu obliczenia wartości dokładnych oraz wektora błędów), a także określenie ilości podprzedziałów, na jakie ma zostać podzielony odcinek  $[a, b]$  (parametr  $n$ , zazwyczaj im większy, tym lepsza dokładność i mniejszy błąd rozwiązania).

Dodatkowo można określić, czy rozwiązania mają zostać naniesione na dwóch oddzielnych wykresach, czy na jednym (parametr `joinPlots`, opis w skrypcie).

## 8 Bibliografia

1. Informacje z wykładu *Metod numerycznych 2* (wydział MiNI PW, dr Iwona Wróbel), w szczególności temat dotyczący *metody Rungego-Kutty* oraz *rozwiązania układu równań różniczkowych*.
2. Z. Fortuna, B. Macukow, J. Wąsowski *Metody numeryczne* - rozdział 7 *Metody rozwiązywania zagadnień początkowych dla równań różniczkowych zwyczajnych*, podrozdział 7.4 *Metody typu Rungego-Kutty*.