

Методы Рунге-Кутта

Залялов Александр, Лобанов Глеб, Солодовников Никита,
группа 183

Постановка задачи

Хотим решить систему

$$\begin{cases} \frac{dy_1}{dt} = f_1(y_1, \dots, y_n) \\ \frac{dy_2}{dt} = f_2(y_1, \dots, y_n) \\ \vdots \\ \frac{dy_n}{dt} = f_n(y_1, \dots, y_n) \end{cases}$$

где y_1, \dots, y_n — функции от t . В векторной форме:

$$\frac{D\vec{y}}{Dt} = f(\vec{y})$$

где $\vec{y} : \mathbb{R} \rightarrow \mathbb{R}^n$, $f : \mathbb{R}^n \rightarrow \mathbb{R}^n$.

Предполагаем известным начальное условие $\vec{y}(t_0) = \vec{y}_0$.

Вид решения

$$\vec{g}^{(i)} = f \left(\vec{y}_0 + h \sum_{j=1}^K a_{ij} \vec{g}^{(j)} \right);$$

$$\vec{y}(t_0 + h) \approx \vec{y}_0 + h \sum_{i=1}^K b_i \vec{g}^{(i)};$$

► K, a_{ij}, b_i — заранее фиксированные константы.

Вид решения

$$\vec{g}^{(i)} = f \left(\vec{y}_0 + h \sum_{j=1}^K a_{ij} \vec{g}^{(j)} \right);$$

$$\vec{y}(t_0 + h) \approx \vec{y}_0 + h \sum_{i=1}^K b_i \vec{g}^{(i)};$$

- ▶ K, a_{ij}, b_i — заранее фиксированные константы.
- ▶ В явных методах $a_{ij} = 0$ при $j \geq i$.

Метод Эйлера

$$K = 1;$$

$$a = (0);$$

$$b = (1);$$

$$K = 4;$$

$$a = \begin{pmatrix} 0 & 0 & 0 & 0 \\ \frac{1}{2} & 0 & 0 & 0 \\ 0 & \frac{1}{2} & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix};$$

$$b = \left(\frac{1}{6} \quad \frac{1}{3} \quad \frac{1}{3} \quad \frac{1}{6} \right).$$

$$\vec{g}^{(1)} = f(\vec{y}_0);$$

$$\vec{g}^{(2)} \approx f(\vec{y}_0) + \frac{h}{2} f'_t(\vec{y}_0);$$

$$\vec{g}^{(3)} \approx f(\vec{y}_0) + \frac{h}{2} f'_t(\vec{y}_0) + \frac{h^2}{4} f''_{tt}(\vec{y}_0);$$

$$\vec{g}^{(4)} \approx f(\vec{y}_0) + hf'_t(\vec{y}_0) + \frac{h^2}{2} f''_{tt}(\vec{y}_0) + \frac{h^3}{4} f'''_{ttt}(\vec{y}_0);$$

$$\vec{g}^{(1)} = f(\vec{y}_0);$$

$$\vec{g}^{(2)} \approx f(\vec{y}_0) + \frac{h}{2} f'_t(\vec{y}_0);$$

$$\vec{g}^{(3)} \approx f(\vec{y}_0) + \frac{h}{2} f'_t(\vec{y}_0) + \frac{h^2}{4} f''_{tt}(\vec{y}_0);$$

$$\vec{g}^{(4)} \approx f(\vec{y}_0) + hf'_t(\vec{y}_0) + \frac{h^2}{2} f''_{tt}(\vec{y}_0) + \frac{h^3}{4} f'''_{ttt}(\vec{y}_0);$$

Линейной комбинацией этого собираем ряд Тейлора для y :

$$y \approx y_0 + hy' + \frac{h^2}{2} y'' + \frac{h^3}{6} y^{(3)} + \frac{h^4}{24} y^{(4)}.$$

Реализация

Для реализации метода Рунге-Кутта мы выбрали язык программирования Haskell, потому что почему бы и нет.

Реализация

Некоторые базовые вещи для удобства:

```
1 class VectorField f where
2     add :: Num a => f a -> f a -> f a
3     dot :: Num a => f a -> f a -> a
4     mul :: Num a => a -> f a -> f a
5
6 instance VectorField [] where
7     add    = zipWith (+)
8     dot a = sum . zipWith (*) a
9     mul x = fmap (* x)
10
```

Реализация

Вся информация о системе хранится в таком типе данных:

```
1 data System = System { getF :: [Double] -> [Double],  
2                        getT0 :: Double,  
3                        getY0 :: [Double]}
```

4

Реализация

Теперь можно реализовать один шаг численного метода.

```
1 stepRK4 :: Double -> System -> System
2 stepRK4 h (System f t0 y0) = System f (t0 + h)
3   (y0 'add' mul (h / 6) g)
4   where
5     g1 = f y0
6     g2 = 2 'mul' f (y0 'add' mul (h / 2) g1)
7     g3 = 2 'mul' f (y0 'add' mul (h / 2) g2)
8     g4 = f (y0 'add' mul h g3)
9     g   = foldr1 add [g1, g2, g3, g4]
10
```

Реализация

Получаем приближённый график функций, решающих систему:

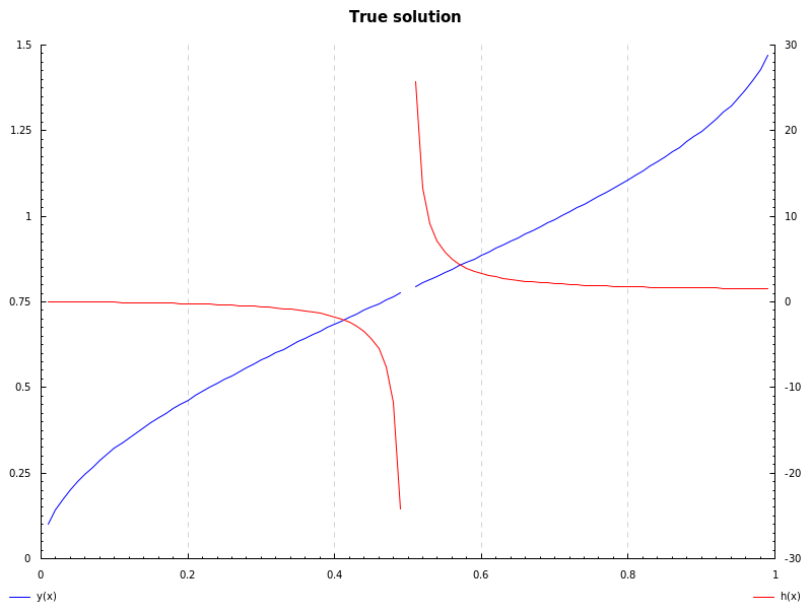
```
1 makeGraph :: (Double, Double) -> Int -> System
2     -> [[Double]]
3 makeGraph interval nSteps system = coords
4     where
5         step    = (snd interval - fst interval)
6                 / fromIntegral nSteps
7         coords = take nSteps
8                 (getY0 <$> iterate (stepRK4 step) system)
9
```

Реализация

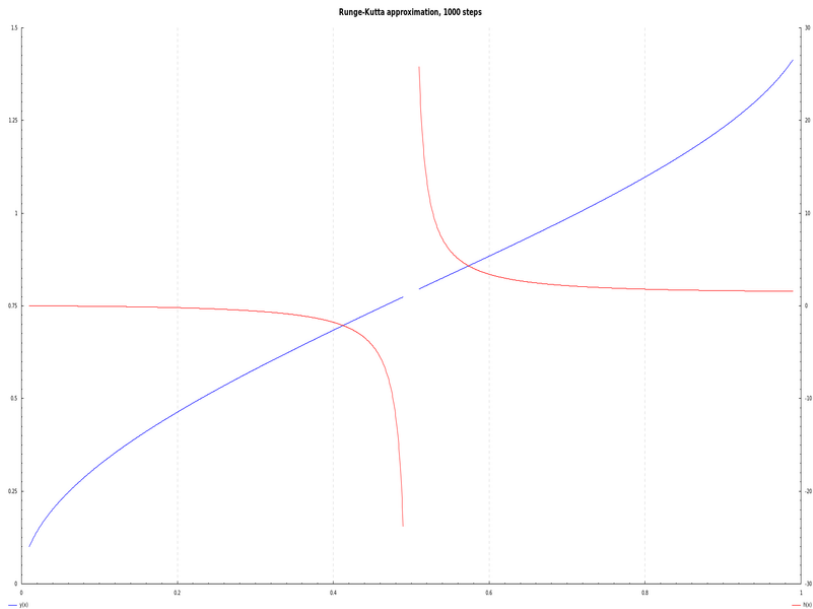
Осталось описать саму систему и запустить код.

```
1 system :: System
2 system = System f t0 (systemSolution t0)
3   where
4     x' _      = 1
5     y' [x, y, h] = 1 / (4 * (x - 0.5) * (h - y))
6     h' [x, y, h] = (y - h) / (x - 0.5)
7     f l         = [x', y', h'] <*> [1]
8
```

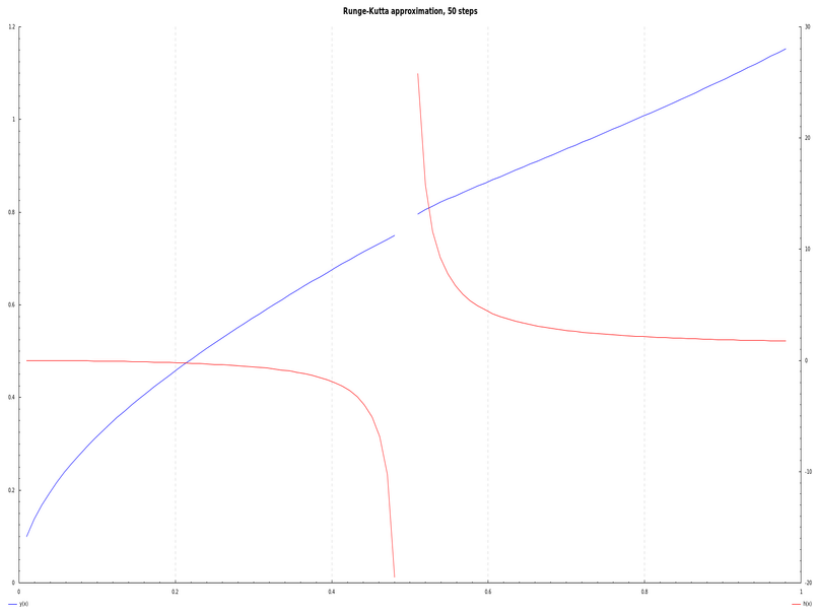
Ответ на систему



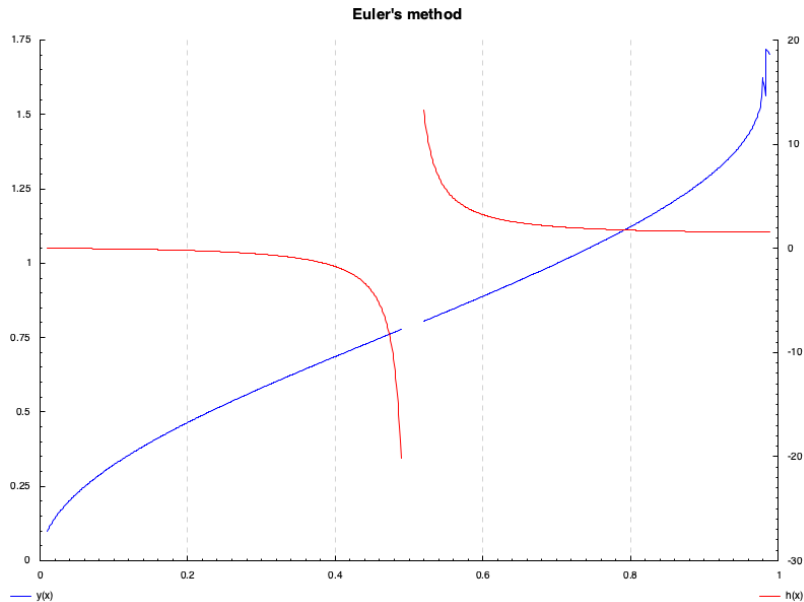
Сравнение методов : Метод Рунге-Кутта при делении отрезка на 1000 частей



Сравнение методов : Метод Рунге-Кутты при делении отрезка на 50 частей



Сравнение методов : Метод Эйлера при делении отрезка на 1000 частей



Анализ графиков

1. Как мы можем видеть метод Рунге-Кутта дает решение почти неотличимое от настоящего при достаточно маленьком размере шага.
2. Метод Эйлера, который является методом 2 порядка, дает хорошие графики только при достаточно маленьком шаге(можно сравнить вывод метода Рунге-Кутта при 50 точках и Эйлера при 1000). Это объясняется тем, что метод Рунге-Кутта имеет больший порядок точности.