Методы Рунге-Кутта

Залялов Александр, Лобанов Глеб, Солодовников Никита,
группа 183

Постановка задачи

Хотим решить систему

$$\begin{cases} \frac{\mathrm{d}y_1}{\mathrm{d}t} = f_1(y_1, \dots, y_n) \\ \frac{\mathrm{d}y_2}{\mathrm{d}t} = f_2(y_1, \dots, y_n) \\ \vdots \\ \frac{\mathrm{d}y_n}{\mathrm{d}t} = f_n(y_1, \dots, y_n) \end{cases}$$

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

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

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

$$K = 1;$$

$$a = \left(0\right);$$

$$b = \left(1\right);$$

RK4

$$K = 4;$$

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

$$b = \begin{pmatrix} \frac{1}{6} & \frac{1}{3} & \frac{1}{3} & \frac{1}{6} \end{pmatrix}.$$

RK4

$$\vec{g}^{(1)} = f(\vec{y_0});$$

$$\vec{g}^{(2)} \approx f(\vec{y_0}) + \frac{h}{2} f'_t(\vec{y_0});$$

$$\vec{g}^{(3)} \approx f(\vec{y_0}) + \frac{h}{2} f'_t(\vec{y_0}) + \frac{h^2}{4} f''_{tt}(\vec{y_0});$$

$$\vec{g}^{(4)} \approx f(\vec{y_0}) + h f'_t(\vec{y_0}) + \frac{h^2}{2} f''_{tt}(\vec{y_0}) + \frac{h^3}{4} f'''_{ttt}(\vec{y_0});$$

RK4

$$\vec{g}^{(1)} = f(\vec{y_0});$$

$$\vec{g}^{(2)} \approx f(\vec{y_0}) + \frac{h}{2} f'_t(\vec{y_0});$$

$$\vec{g}^{(3)} \approx f(\vec{y_0}) + \frac{h}{2} f'_t(\vec{y_0}) + \frac{h^2}{4} f''_{tt}(\vec{y_0});$$

$$\vec{g}^{(4)} \approx f(\vec{y_0}) + h f'_t(\vec{y_0}) + \frac{h^2}{2} f''_{tt}(\vec{y_0}) + \frac{h^3}{4} f'''_{ttt}(\vec{y_0});$$

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:

```
class VectorField f where
    add :: Num a => f a -> f a -> f a
    dot :: Num a => f a -> f a -> a
    mul :: Num a => a -> f a -> f a

instance VectorField [] where
    add   = zipWith (+)
    dot a = sum . zipWith (*) a
    mul x = fmap (* x)
```

Реализация

Вся информация о системе хранится в таком типе данных:

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

Реализация

Теперь можно реализовать один шаг численного метода.

```
stepRK4 :: Double -> System -> System
stepRK4 h (System f t0 y0) = System f (t0 + h)
    (y0 `add` mul (h / 6) g)
  where
    g1 = f y0
    g2 = 2 `mul` f (y0 `add` mul (h / 2) g1)
    g3 = 2 `mul` f (y0 `add` mul (h / 2) g2)
    g4 = f (y0 `add` mul h g3)
    g  = foldr1 add [g1, g2, g3, g4]
```

Реализация

Получаем приближённый график функций, решающих систему:

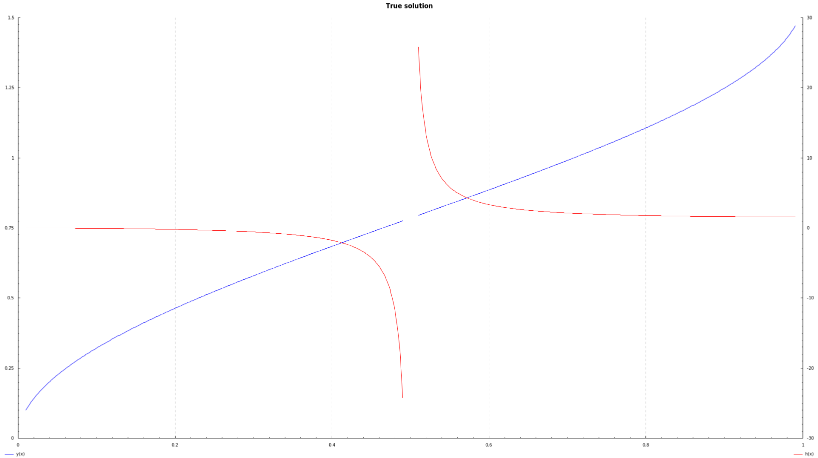
```
makeGraph :: (Double, Double) -> Int -> System
    -> [[Double]]
makeGraph interval nSteps system = coords
  where
    step   = (snd interval - fst interval)
        / fromIntegral nSteps
    coords = take nSteps
        (getY0 <$> iterate (stepRK4 step) system)

```

Реализация

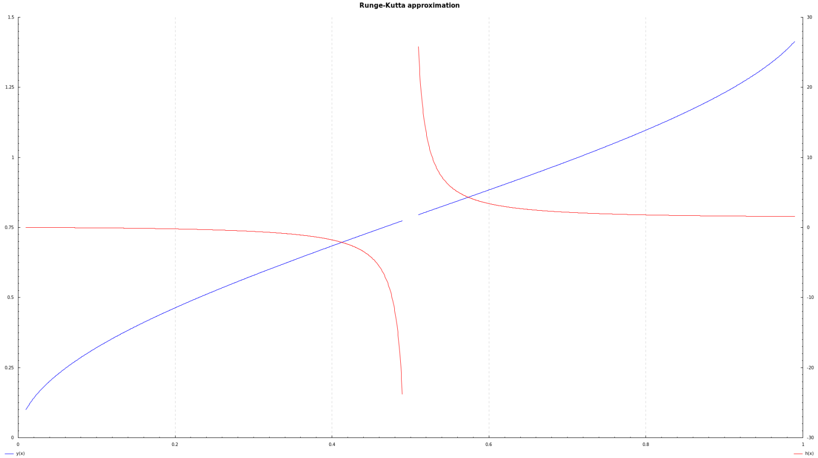
Осталось описать саму систему и запустить код.

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

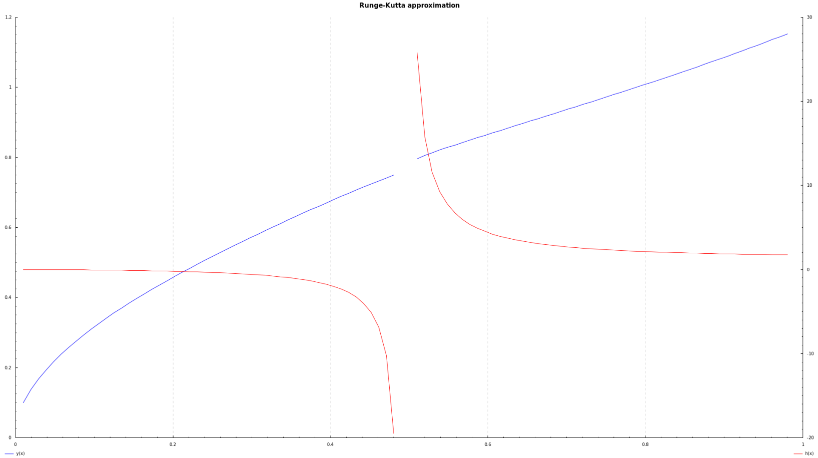
Ответ на систему



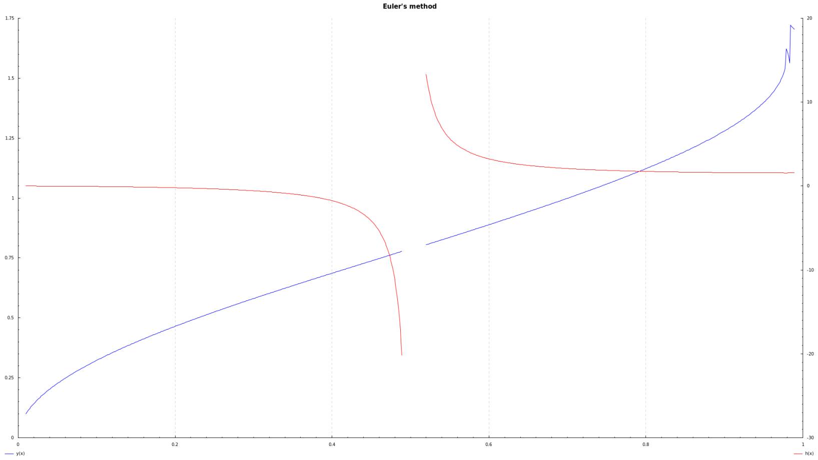
Сравнение методов : Метод Рунге-Кутта при делении отрезка на 1000 частей



Сравнение методов : Метод Рунге-Кутта 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 1 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.