

Презентация к лабораторной работе №7

Боровикова Карина Владимировна

2023, 25 марта

RUDN University, Moscow, Russian Federation

- Важность умения работы с языками Julia и OpenModelica в части математического моделирования

- Язык Julia
- Язык OpenModelica
- Моделирование задачи о распространении рекламы

Цель работы:

Построить модель для задачи о распространении рекламы с помощью языков Julia и OpenModelica для трех случаев

Задачи:

- Построить графики распространения рекламы, математическая модель которой описывается уравнениями в задании к лабораторной работе №7 для трех случаев при $N = 1224$, $n = 14$
- Для случая 2 определить, в какой момент времени скорость распространения рекламы будет иметь максимальное значение.

1. Рассматриваем задачу о распространении рекламы (рис. 1).

Вариант № 18

Постройте график распространения рекламы, математическая модель которой описывается следующим уравнением:

$$1. \quad \frac{dn}{dt} = (0.61 + 0.000061n(t))(N - n(t))$$

$$2. \quad \frac{dn}{dt} = (0.000073 + 0.73n(t))(N - n(t))$$

$$3. \quad \frac{dn}{dt} = (0.7t + 0.6\cos(t)n(t))(N - n(t))$$

При этом объем аудитории $N = 1224$, в начальный момент о товаре знает 14 человек. Для случая 2 определите в какой момент времени скорость распространения рекламы будет иметь максимальное значение.

Рис. 1: Моделирование распространения рекламы

2. Программируем первый случай на Julia (рис. 2).

"C:\\Users\\kvbk2\\.julia\\pluto_notebooks\\lab07_julia_1.png"

```
begin
  using Plots ✓
  using DifferentialEquations ✓

  a = 0.61
  b = 0.000061
  N = 1224
  n0 = 14
  t = (0.0, 5.0)

  function F!(dn, n, p, t)
    dn[1] = (a+b*n[1])*(N-n[1])
  end

  prob = ODEProblem(F!, [n0], t)
  sol = solve(prob, saveat = 0.005)

  plt = plot(
    dpi = 300,
    size = (800, 600),
    title = "Модель эффективности рекламы"
  )

  plot!(
    sol,
    color = :red,
    xlabel = "t",
    ylabel = "Распространение рекламы",
    label = "Информированные клиенты"
  )
  #plot(sol)
```

3. Программируем первый случай на OpenModelica (рис. 3).

```
1  model lab071
2      constant Real a = 0.61;
3      constant Real b = 0.000061;
4      constant Real N = 1224;
5      Real n(start=14);
6
7  equation
8      der(n) = (a+b*n)*(N-n);
9
10 end lab071;
```

Рис. 3: Первый случай на OpenModelica

4. Программируем второй случай на Julia (рис. 4).

```
begin
    using Plots ✓
    using DifferentialEquations ✓

    a = 0.000073
    b = 0.73
    N = 1224
    n0 = 14
    t = (0.0, 0.05)

    max = [0.0, 0.0, 0.0]

    function F!(dn, n, p, t)

        dn[1] = (a+b*n[1])*(N-n[1])

        if dn[1] > max[1]
            max[1] = dn[1]
            max[2] = n[1]
            max[3] = t
        end
    end

    prob = ODEProblem(F!, [n0], t)
    sol = solve(prob, saveat = 0.000001)

    plt = plot(
        dpi = 300,
        size = (800, 600),
        title = "Модель эффективности рекламы"
    )

    plot!(
        sol,
        color =:red,
        xlabel = "t",
        ylabel = "Распространение рекламы",
        label = "Информированные клиенты"
    )

    scatter!(
        plt,
        [max[3]],
        [max[2]],
        label = "Максимальная скорость",
    )
end
```


5. Програмируем второй случай на OpenModelica (рис. 5).

```
1  model lab072
2      constant Real a = 0.000073;
3      constant Real b = 0.073;
4      constant Real N = 1224;
5      Real n(start=14);
6
7      equation
8          der(n) = (a+b*n)*(N-n);
9
10     end lab072;
11     |
```

Рис. 5: Второй случай на OpenModelica

6. Программируем третий случай на Julia (рис. 6).

```
begin
    using Plots ✓
    using DifferentialEquations ✓

    a = 0.7
    b = 0.6
    N = 1224
    n0 = 14
    t = (0.0, 0.05)

    max = [0.0, 0.0, 0.0]

    function F!(dn, n, p, t)

        dn[1] = (a*t+b*cos(t))*n[1]*(N-n[1])

    end

    prob = ODEProblem(F!, [n0], t)
    sol = solve(prob, saveat = 0.000001)

    plt = plot(
        dpi = 300,
        size = (800, 600),
        title = "Модель эффективности рекламы"
    )
    plot!(
        sol,
        color = :red,
        xlabel = "t",
        ylabel = "Распространение рекламы",
        label = "Информированные клиенты"
    )

    savefig(plt, "lab07_julia_3.png")
```

7. Програмируем третий случай на OpenModelica (рис. 7).

```
1  model lab073
2      constant Real a = 0.7;
3      constant Real b = 0.6;
4      constant Real N = 1224;
5      Real n(start=14);
6
7  equation
8      der(n) = (a*time+b*cos(time)*n)*(N-n);
9
10 end lab073;
11
```

Рис. 7: Третий случай на OpenModelica

6. Запускаем код через терминал, получаем изображения для случаев на Julia (рис. 8).

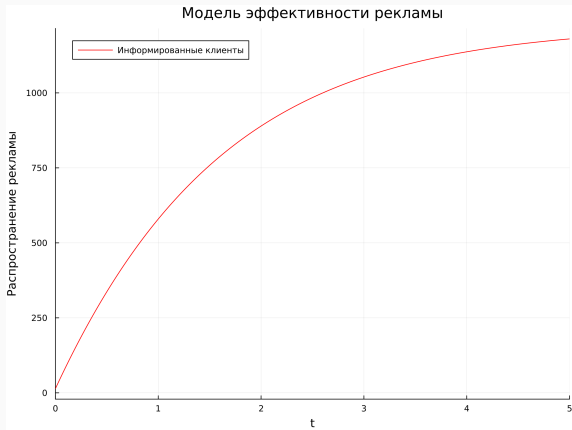


Рис. 8: Результаты выполнения кодов на Julia

7. Запускаем код через терминал, получаем изображения случаев на Julia (рис. 9).

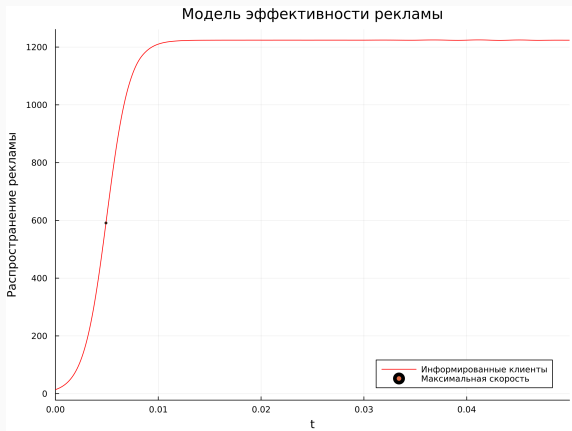


Рис. 9: Результаты выполнения кодов на Julia

8. Запускаем код через терминал, получаем изображения случаев на Julia (рис. 10).

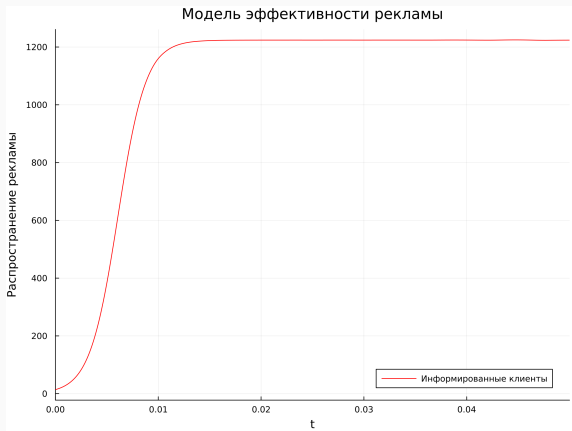


Рис. 10: Результаты выполнения кодов на Julia

9. Запускаем код через терминал, получаем изображения для случаев на OpenModelica (рис. 11).

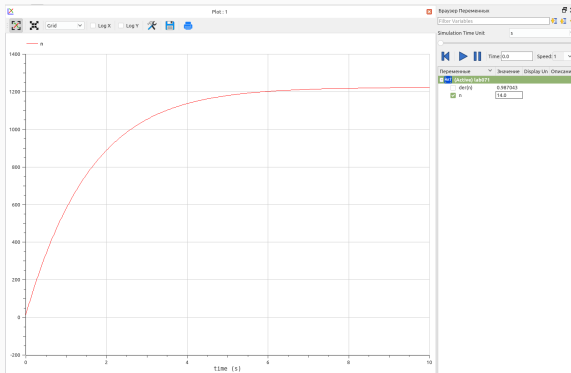


Рис. 11: Результаты выполнения кодов на Julia

9. Запускаем код через терминал, получаем изображения для случаев на OpenModelica (рис. 12).

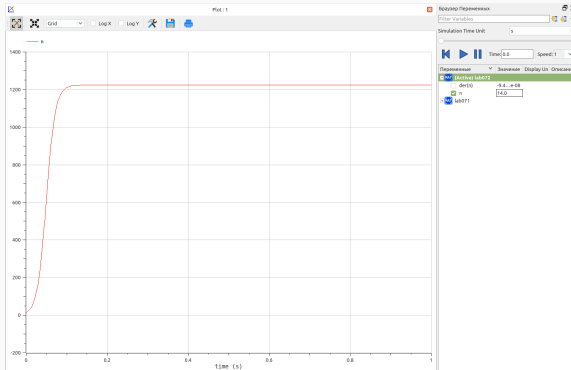


Рис. 12: Результаты выполнения кодов на Julia

10. Запускаем код через терминал, получаем изображения для

Результаты

В ходе выполнения лабораторной работы мне удалось построить модель для задачи о распространении рекламы с помощью языков Julia и OpenModelica в трех случаях