Отчет по лабораторной работе №7

Дисциплина: Математическое моделирование

Выполнила: Губина Ольга Вячеславовна

Содержание

1 Цель работы **5**

2 Задание **6**

3 Теоретическое введение **7**

4 Выполнение лабораторной работы **9**

4.1 Первый случай 9
4.2 Второй случай 13
4.3 Треий случай 18

5 Выводы **23**

Список литературы **24**

Список иллюстраций

3.1 График решения уравнения модели Мальтуса 8
3.2 График логистической кривой 8

4.1 Распространение рекламы случай 1 - Julia 12
4.2 Распространение рекламы случай 1 - OpenModelica 13
4.3 Распространение рекламы случай 2 - Julia 17
4.4 Распространение рекламы случай 2 - OpenModelica 18
4.5 Распространение рекламы случай 3 - Julia 21
4.6 Распространение рекламы случай 3 - OpenModelica 22

Список таблиц

1 Цель работы

Смоделировать распространение рекламы по средством языков программирования Julia и OpenModelica.

2 Задание

- Постройте график распространения рекламы для трех случаев
- Для случая 2 определите в какой момент времени скорость распространения рекламы будет иметь максимальное значение

3 Теоретическое введение

Модель рекламной кампании описывается следующими величинами. Считаем, что $\frac{dn}{dt}$ - скорость изменения со временем числа потребителей, узнавших о товаре и готовых его купить, t - время, прошедшее с начала рекламной кампании, $n(t)$ - число уже информированных клиентов. Эта величина пропорциональна числу покупателей, еще не знающих о нем, это описывается следующим образом: $\alpha_1(t)(N-n(t))$, где N - общее число потенциальных платежеспособных покупателей, $\alpha_1(t)>0$ - характеризует интенсивность рекламной кампании (зависит от затрат на рекламу в данный момент времени). Помимо этого, узнавшие о товаре потребители также распространяют полученную информацию среди потенциальных покупателей, не знающих о нем (в этом случае работает т.н. сарафанное радио). Этот вклад в рекламу описывается величиной $\alpha_2(t)n(t)(N-n(t))$, эта величина увеличивается с увеличением потребителей узнавших о товаре. Математическая модель распространения рекламы описывается уравнением (формула [3.1])[1]:

$$\frac{dn}{dt} = (\alpha_1(t) + \alpha_2(t)n(t))(N - n(t)) \tag{3.1}$$

При $\alpha_1(t) >> \alpha_2(t)$ получается модель типа модели Мальтуса (также именуется как мальтузианская модель - экспоненциальный рост с постоянным темпом [2]), решение которой имеет вид (рис. [3.1])

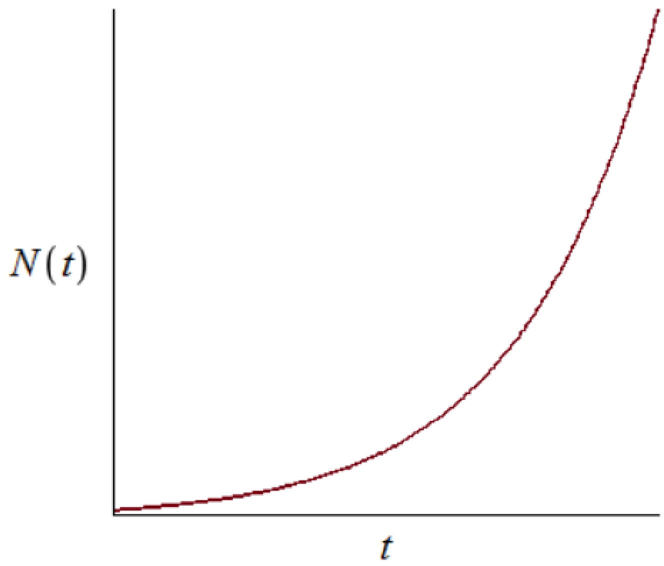

Рис. 3.1: График решения уравнения модели Мальтуса

В обратном случае, при $\alpha_1(t) << \alpha_2(t)$ получаем уравнение логистической кривой (кривая, показывающая поведение с течением времени переменной , где $a < x < b$, и увеличение x задается формулой $\frac{dx}{dt} = \alpha(x - a)(b – x)$ [3]) (рис. [3.2]):

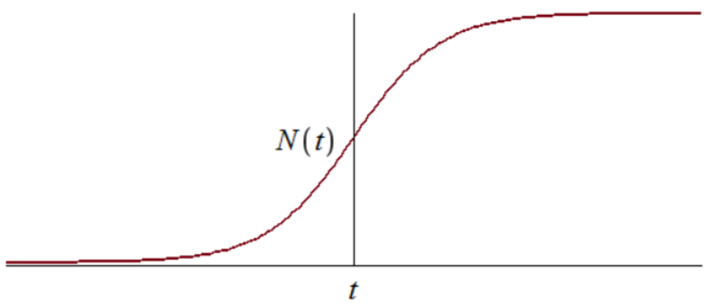

Рис. 3.2: График логистической кривой

4 Выполнение лабораторной работы

1. К выполнению нам предлагается выполнить соответстующий номеру студенчесткого билета вариант: 1032201737 % 70 + 1 = 8

2. Задача предложенного варианта состоит в следующем:

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

$$\frac{dn}{dt} = (0.64 + 0.00014n(t))(N - n(t))$$

$$\frac{dn}{dt} = (0.000014 + 0.63n(t))(N - n(t))$$

$$\frac{dn}{dt} = (0.7t + 0.4\cos t n(t))(N - n(t))$$

При этом объем аудитории $N = 810$, в начальный момент о товаре знает 11 человек. Для случая 2 определите в какой момент времени скорость распространения рекламы будет иметь максимальное значение.

4.1 Первый случай

1. Для первого случая распространение рекламы описывается следующим уравнением:

$$\frac{dn}{dt} = (0.64 + 0.00014n(t))(N - n(t)) \tag{4.1}$$

Отсюда видно, что:

$$\alpha_1(t) = 0.64$$

$$\alpha_2(t) = 0.00014$$

Это значит, что

$$\alpha_1(t) >> \alpha_2(t)$$

, значит, должна получиться модель типа Мальтуса.

На языке Julia напишем код моделирующий распространение рекламы:

```
using Plots
using DifferentialEquations


"Условия:"
N = 810


n_0 = 11


u_0 = [n_0]
T = (0.0, 50.0) # отслеживаемый промежуток времени


function a_1(t)
    return 0.64
end


function a_2(t)
```

```
    return 0.00014
end

function F!(du, u, p, t)
    du[1] = (a_1(t) + a_2(t) * u[1]) * (N - u[1])
end

prob = ODEProblem(F!, u_0, T)
sol = solve(prob, saveat = 0.0001) # обозначили шаг

const NN = Float64[]

for u in sol.u
    n = u[1]
    push!(NN, n)
end

plt = plot(
    dpi = 300,
    size = (800, 600),
    title = "Модель распространения рекламы - случай 1"
)

plot!(
    plt,
    sol.t,
    NN,
    color = :blue,
    xlabel="t",
```

```
    ylabel="N(t)",
    label = "Число осведомленных"
)

savefig(plt, "julia_1.png")
```

В качестве результата у нас график распространения рекламы (рис. [4.1]):

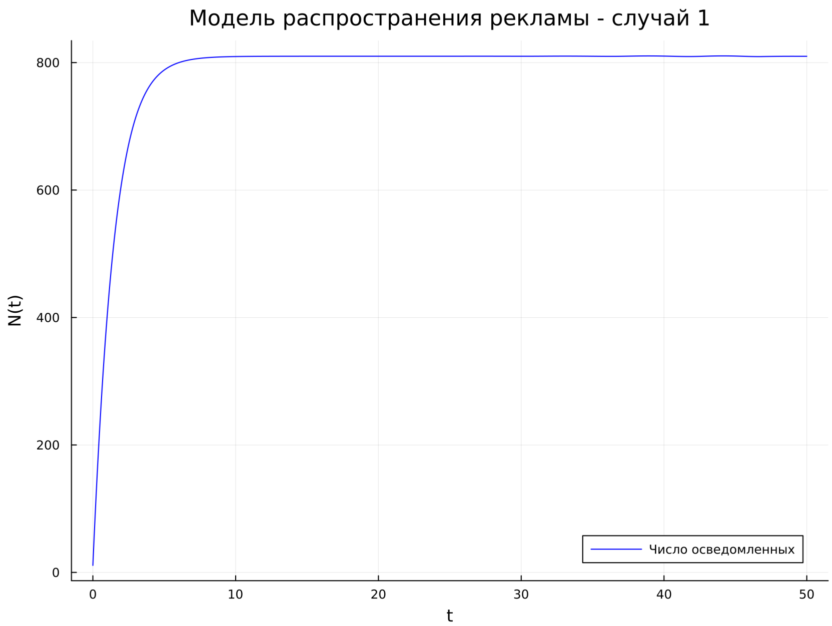

Рис. 4.1: Распространение рекламы случай 1 - Julia

2. Напишем код на OpenModelica:

```
model lab07_1
  constant Integer N = 810;
  constant Integer n_0 = 11;
  constant Real a_1 = 0.64;
  constant Real a_2 = 0.00014;
  Real n(start=n_0);
  Real t = time;
```

```
equation
  der(n) = (a_1+a_2*n)*(N-n);
  annotation(experiment(StartTime = 0, StopTime = 50.0),
    Documentation);
end lab07_1;
```

В качестве результата у нас график изменения численности численности заболеваемости (рис. [4.2]):

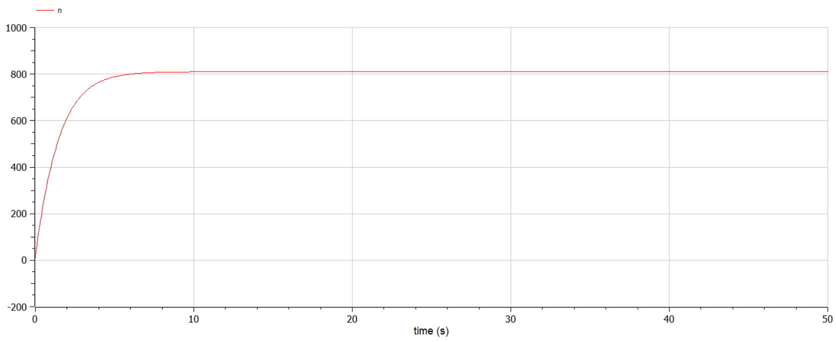

Рис. 4.2: Распространение рекламы случай 1 - OpenModelica

4.2 Второй случай

1. Для второго случая распространение рекламы описывается следующим уравнением:

$$\frac{dn}{dt} = (0.000014 + 0.63n(t))(N - n(t)) \tag{4.2}$$

Отсюда видно, что:

- $$\alpha_1(t) = 0.000014$$

- $$\alpha_2(t) = 0.63$$

Это значит, что

$$\alpha_1(t) << \alpha_2(t)$$

, а значит, график будет иметь вид логистической кривой.

Также найдем точку, в которой скорость распространения рекламы будет максимальной.

На языке Julia напишем код моделирующий распространение рекламы:

```
using Plots
using DifferentialEquations


"Условия:"
N = 810


n_0 = 11


u_0 = [n_0]
T = (0.0, 0.5) # отслеживаемый промежуток времени


max_v = [0.0, 0.0, 0.0] # для поиска максимальной скорости [скорость, кол-
во пользователей, время]


function a_1(t)
    return 0.000014
end


function a_2(t)
    return 0.63
end
```

```
function F!(du, u, p, t)
    du[1] = (a_1(t) + a_2(t) * u[1]) * (N - u[1])

    if du[1] > max_v[1]
        max_v[1] = du[1]
        max_v[2] = u[1]
        max_v[3] = t
    end
end

prob = ODEProblem(F!, u_0, T)
sol = solve(prob, saveat = 0.0001) # обозначили шаг

@show max_v[3]

const NN = Float64[]

for u in sol.u
    n = u[1]
    push!(NN, n)
end

plt = plot(
    dpi = 300,
    size = (800, 600),
    title = "Модель распространения рекламы - случай 2"
)

plot!(
```

```
    plt,
    sol.t,
    NN,
    color = :blue,
    xlabel="t",
    ylabel="N(t)",
    label = "Число осведомленных"
)

scatter!(
    plt,
    [max_v[3]],
    [max_v[2]],
    label="Момент максимальной скорости",
    ms=1.5
)

savefig(plt, "julia_2.png")
```

В качестве результата у нас график распространения рекламы (рис. [4.3]):

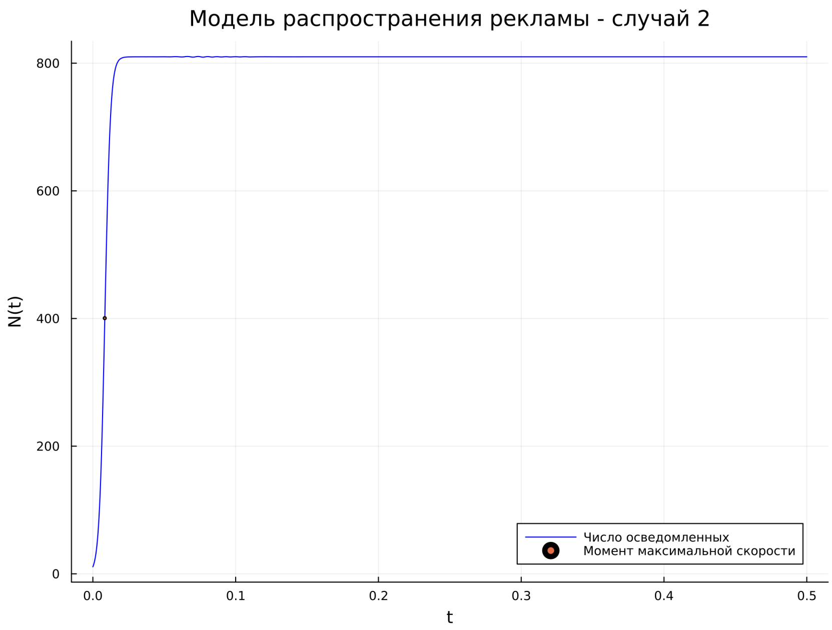

Рис. 4.3: Распространение рекламы случай 2 - Julia

Момент времени максимальной скорости распространения (рис. [??]):

```
PS D:\2022-2023\Математическое моделирование\mathmod\labs\lab07\progs> julia lab07_2.jl
max_v[3] = 0.008318709157747696
PS D:\2022-2023\Математическое моделирование\mathmod\labs\lab07\progs>
```

2. Напишем код на OpenModelica:

```
model lab07_2
  constant Integer N = 810;
  constant Integer n_0 = 11;
  constant Real a_1 = 0.000014;
  constant Real a_2 = 0.63;
  Real n(start=n_0);
  Real t = time;
equation
  der(n) = (a_1+a_2*n)*(N-n);
  annotation(experiment(StartTime = 0.0, StopTime = 0.5),
    Documentation);
end lab07_2;
```

В качестве результата у нас график изменения численности численности заболеваемости (рис. [4.4]):

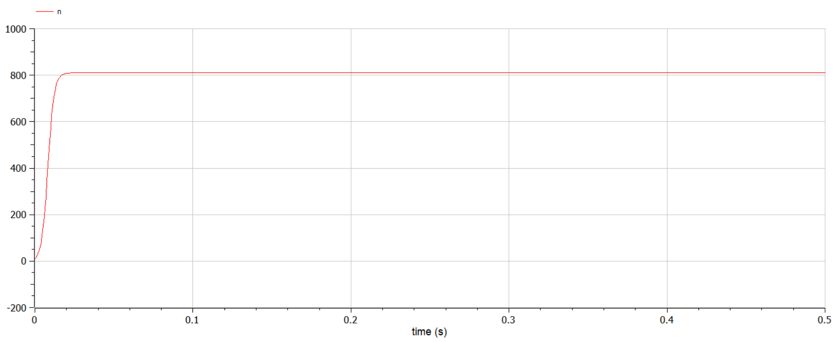

Рис. 4.4: Распространение рекламы случай 2 - OpenModelica

4.3 Треий случай

1. Для первого случая распространение рекламы описывается следующим уравнением:

$$\frac{dn}{dt} = (0.7t + 0.4\cos tn(t))(N - n(t)) \tag{4.3}$$

Отсюда видно, что:

- $$\alpha_1(t) = 0.7t$$

- $$\alpha_2(t) = 0.4\cos t$$

На языке Julia напишем код моделирующий распространение рекламы:

```
using Plots
using DifferentialEquations
```

```
"Условия:"
N = 810

n_0 = 11

u_0 = [n_0]
T = (0.0, 0.5) # отслеживаемый промежуток времени

function a_1(t)
    return 0.7*t
end

function a_2(t)
    return 0.4*cos(t)
end

function F!(du, u, p, t)
    du[1] = (a_1(t) + a_2(t) * u[1]) * (N - u[1])
end

prob = ODEProblem(F!, u_0, T)
sol = solve(prob, saveat = 0.0001) # обозначили шаг

const NN = Float64[]

for u in sol.u
    n = u[1]
    push!(NN, n)
```

```
end

plt = plot(
    dpi = 300,
    size = (800, 600),
    title = "Модель распространения рекламы - случай 1"
)

plot!(
    plt,
    sol.t,
    NN,
    color = :blue,
    xlabel="t",
    ylabel="N(t)",
    label = "Число осведомленных"
)

savefig(plt, "julia_3.png")
```

В качестве результата у нас график распространения рекламы (рис. [4.5]):

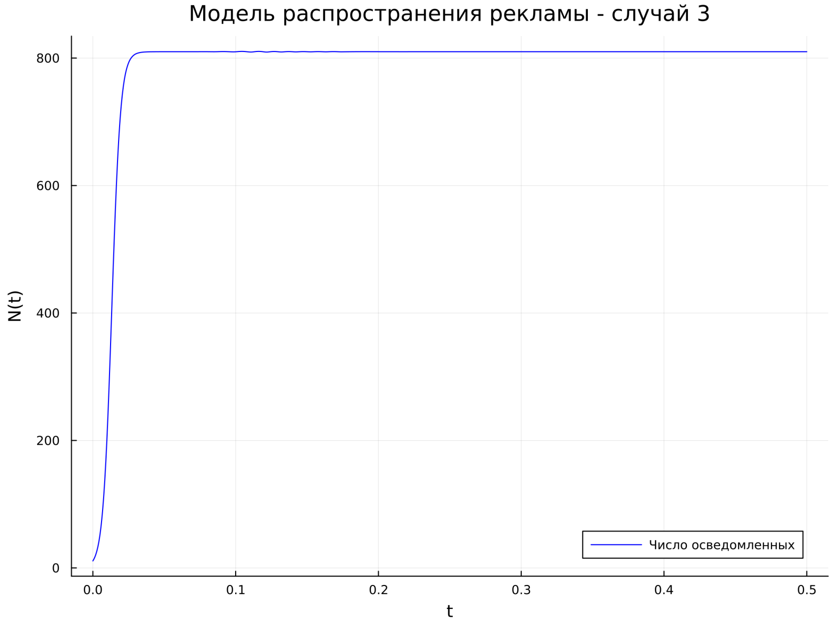

Рис. 4.5: Распространение рекламы случай 3 - Julia

2. Напишем код на OpenModelica:

```
model lab07_3
  constant Integer N = 810;
  constant Integer n_0 = 11;
  Real n(start=n_0);
  Real t = time;
equation
  der(n) = (0.7*t+0.4*cos(t)*n)*(N-n);
  annotation(experiment(StartTime = 0, StopTime = 0.5),
    Documentation);
end lab07_3;
```

В качестве результата у нас график изменения численности численности заболеваемости (рис. [4.6]):

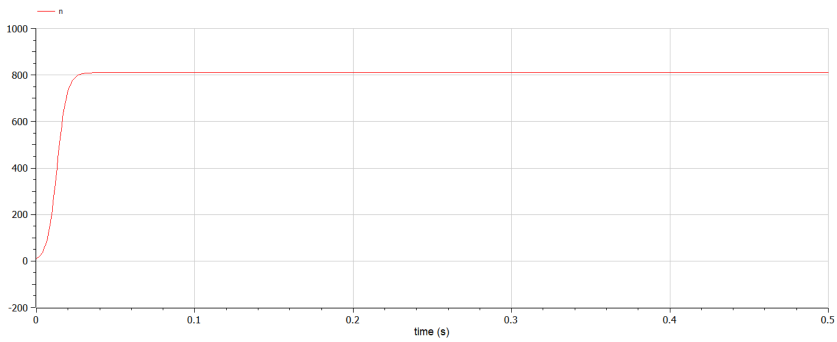

Рис. 4.6: Распространение рекламы случай 3 - OpenModelica

5 Выводы

- Построила график распространения рекламы для трех случаев
- Для случая 2 определила в какой момент времени скорость распространения рекламы будет иметь максимальное значение

Список литературы

1. Эффективность рекламы [Электронный ресурс]. 2023. URL: https://esystem.rudn.ru/pluginfile.php/1971741/mod_resource/content/2/%D0%9B%D0%B0%D0%B1%D0%BE%D1%80%D0%B0%D1%82%D0%BE%D1%80%D0%BD%D0%B0%D1%8F%20%D1%80%D0%B0%D0%B1%D0%BE%D1%82%D0%B0%20%E2%84%96%206.pdf.
2. Мальтузианская модель роста [Электронный ресурс]. 2022. URL: https://ru.wikipedia.org/wiki/%D0%9C%D0%B0%D0%BB%D1%8C%D1%82%D1%83%D0%B7%D0%B8%D0%B0%D0%BD%D1%81%D0%BA%D0%B0%D1%8F_%D0%BC%D0%BE%D0%B4%D0%B5%D0%BB%D1%8C_%D1%80%D0%BE%D1%81%D1%82%D0%B0.
3. ЛОГИСТИЧЕСКАЯ КРИВАЯ [Электронный ресурс]. URL: https://ecolog.academic.ru/4608/%D0%9B%D0%9E%D0%93%D0%98%D0%A1%D0%A2%D0%98%D0%A7%D0%95%D0%A1%D0%9A%D0%90%D0%AF.