

# Лабораторная работа №7

## Модель распространения рекламы

---

Губина О. В.

25 марта 2023

Российский университет дружбы народов, Москва, Россия

## Информация

---

- Губина Ольга Вячеславовна
- студент(-ка) уч. группы НПИбд-01-20
- Российский университет дружбы народов
- 1032201737@pfur.ru
- <https://github.com/ovgubina>

## Вводная часть

---

- Необходимость навыков моделирования реальных математических задач, построение графиков.

- Простейшая модель распространения рекламы
- Языки для моделирования:
  - Julia
  - OpenModelica

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

- Языки для моделирования:
  - Julia
  - OpenModelica



## Процесс выполнения работы

---

Математическая модель распространения рекламы описывается уравнением:

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

При  $\alpha_1(t) \gg \alpha_2(t)$  получается модель типа модели Мальтуса, в обратном случае, при  $\alpha_1(t) \ll \alpha_2(t)$  получаем уравнение логистической кривой.

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

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

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

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

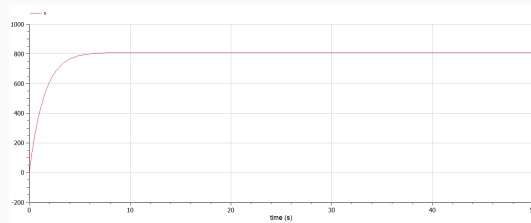
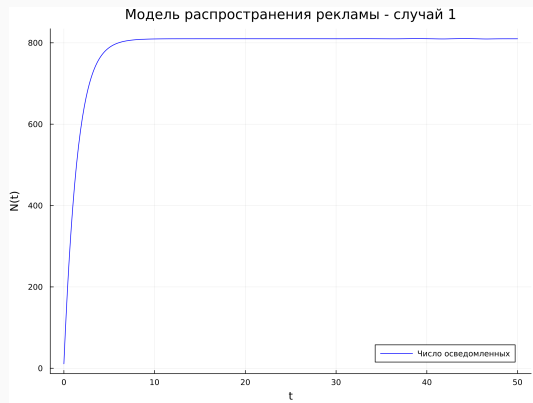
При этом объем аудитории  $N = 810$ , в начальный момент о товаре знает 11 человек. Для случая 2 определите в какой момент времени скорость распространения рекламы будет иметь максимальное значение.

# Первый случай - Julia

```
1 using Plots
2 using DifferentialEquations
3
4 "Условия:"
5 N = 810
6
7 n_0 = 11
8
9 u_0 = [n_0]
10 T = (0.0, 50.0) # отслеживаемый промежуток времени
11
12 function a_1(t)
13     return 0.64
14 end
15
16 function a_2(t)
17     return 0.00014
18 end
19
20 function F!(du, u, p, t)
21     du[1] = (a_1(t) + a_2(t) * u[1]) * (N - u[1])
22 end
23
24 prob = ODEProblem(F!, u_0, T)
25 sol = solve(prob, saveat = 0.0001) # обозначили шаг
26
27 const NN = Float64[]
28
29 for u in sol.u
30     n = u[1]
31     push!(NN, n)
32 end
33
34 plt = plot(
35     dpi = 300,
36     size = (800, 600),
37     title = "Модель распространения рекламы - случай 1"
38 )
39
40 plot!(
41     plt,
42     sol.t,
43     NN,
44     color = :blue,
45     xlabel = "t",
46     ylabel = "N(t)",
47     label = "Число осведомленных"
48 )
49
50 savefig(plt, "julia_1.png")
```

```
1 model lab07_1
2   constant Integer N = 810;
3   constant Integer n_0 = 11;
4   constant Real a_1 = 0.64;
5   constant Real a_2 = 0.00014;
6   Real n(start=n_0);
7   Real t = time;
8 equation
9   der(n) = (a_1+a_2*n)*(N-n);
10  annotation(experiment(StartTime = 0, StopTime = 50.0),
11  Documentation);
12 end lab07_1;
13
```

# Первый случай - графики



## Второй случай - код на Julia

```
1 using Plots
2 using DifferentialEquations
3
4 "Условия:"
5 N = 810
6
7 n_0 = 11
8
9 u_0 = [n_0]
10 T = (0.0, 0.5) # отслеживаемый промежуток времени
11
12 max_v = [0.0, 0.0, 0.0] # для поиска максимальной скорости [скорость, кол-во пользователей, время]
13
14 function a_1(t)
15     return 0.000014
16 end
17
18 function a_2(t)
19     return 0.63
20 end
21
22 function F!(du, u, p, t)
23     du[1] = (a_1(t) + a_2(t) * u[1]) * (N - u[1])
24
25     if du[1] > max_v[1]
26         max_v[1] = du[1]
27         max_v[2] = u[1]
28         max_v[3] = t
29     end
30 end
31
32 prob = ODEProblem(F!, u_0, T)
33 sol = solve(prob, saveat = 0.0001) # обозначили шаг
34
```

```
35 @show max_v[3]
36
37 const NN = Float64[]
38
39 for u in sol.u
40     n = u[1]
41     push!(NN, n)
42 end
43
44 plt = plot(
45     dpi = 300,
46     size = (800, 600),
47     title = "Модель распространения рекламы - случай 2"
48 )
49
50 plot!(
51     plt,
52     sol.t,
53     NN,
54     color = :blue,
55     xlabel="t",
56     ylabel="N(t)",
57     label = "Число осведомленных"
58 )
59
60 scatter!(
61     plt,
62     [max_v[3]],
63     [max_v[2]],
64     label="Момент максимальной скорости",
65     ms=1.5
66 )
67
68 savefig(plt, "julia_2.png")
```

## Второй случай - момент времени с максимальной скоростью распространения

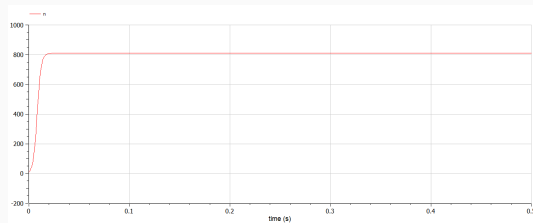
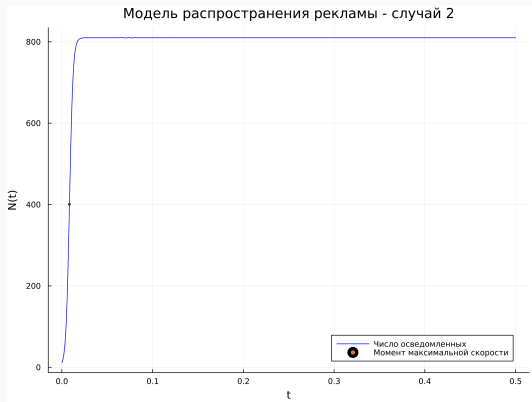
```
PS D:\2022-2023\Математическое моделирование\mathmod\labs\lab07\progs> julia lab07_2.jl  
max_v[3] = 0.008318709157747696  
PS D:\2022-2023\Математическое моделирование\mathmod\labs\lab07\progs>
```



## Второй случай - код на OpenModelica

```
1 model lab07_2
2   constant Integer N = 810;
3   constant Integer n_0 = 11;
4   constant Real a_1 = 0.000014;
5   constant Real a_2 = 0.63;
6   Real n(start=n_0);
7   Real t = time;
8   equation
9     der(n) = (a_1+a_2*n)*(N-n);
10  annotation(experiment(StartTime = 0.0, StopTime = 0.5),
11  Documentation);
12 end lab07_2;
13
```

## Второй случай - графики

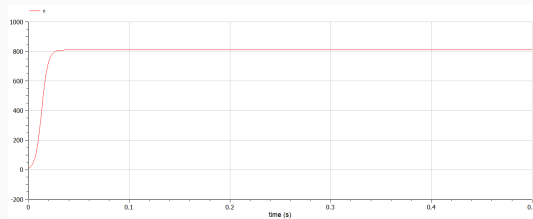
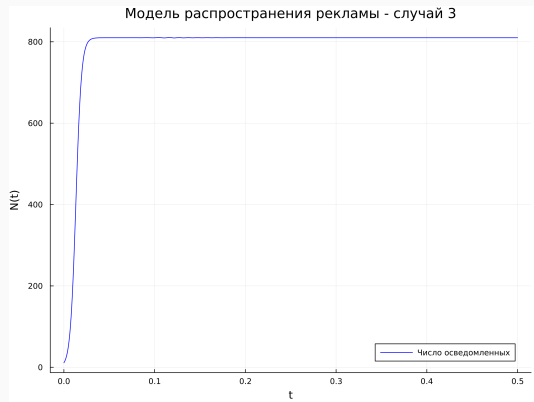


## Третий случай - Julia

```
1 using Plots
2 using DifferentialEquations
3
4 "Условия:"
5 N = 810
6
7 n_0 = 11
8
9 u_0 = [n_0]
10 T = (0.0, 0.5) # отслеживаемый промежуток времени
11
12 function a_1(t)
13     return 0.7*t
14 end
15
16 function a_2(t)
17     return 0.4*cos(t)
18 end
19
20 function F!(du, u, p, t)
21     du[1] = (a_1(t) + a_2(t) * u[1]) * (N - u[1])
22 end
23
24 prob = ODEProblem(F!, u_0, T)
25 sol = solve(prob, saveat = 0.0001) # обозначили шаг
26
27 const NN = Float64[]
28
29 for u in sol.u
30     n = u[1]
31     push!(NN, n)
32 end
33
34 plt = plot(
35     dpi = 300,
36     size = (800, 600),
37     title = "Модель распространения рекламы - случай 1"
38 )
39
40 plot!(
41     plt,
42     sol.t,
43     NN,
44     color = :blue,
45     xlabel="t",
46     ylabel="N(t)",
47     label = "Число осведомленных"
48 )
49
50 savefig(plt, "julia_3.png")
```

```
1 model lab07_3
2   constant Integer N = 810;
3   constant Integer n_0 = 11;
4   Real n(start=n_0);
5   Real t = time;
6   equation
7     der(n) = (0.7*t+0.4*cos(t)*n)*(N-n);
8   annotation(experiment(StartTime = 0, StopTime = 0.5),
9     Documentation);
10 end lab07_3;
11
```

## Третий случай - графики



## Результаты работы

---

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

## Вывод

---



Смоделировала распространение рекламы по средством языков программирования Julia и OpenModelica.