

Лабораторная работа №7

Эффективность рекламы

Егорова Д.В.

22 марта 2023

Российский университет дружбы народов, Москва, Россия

Информация

- Егорова Диана Витальевна
- студент НФИбд-01-20
- Российский университет дружбы народов
- 1032201662@rudn.ru

Вводная часть

- Моделирование ситуации
- Наглядное представление
- Простота использования

- Рассмотреть задачу об эффективности рекламы
- Построить графики распространения рекламы, математическая модель которой описывается одним из трех уравнений
- Рассмотреть, как будет протекать эффективность рекламы в трех случаях
- Для одного случая определить в какой момент времени скорость распространения рекламы будет иметь максимальное значение

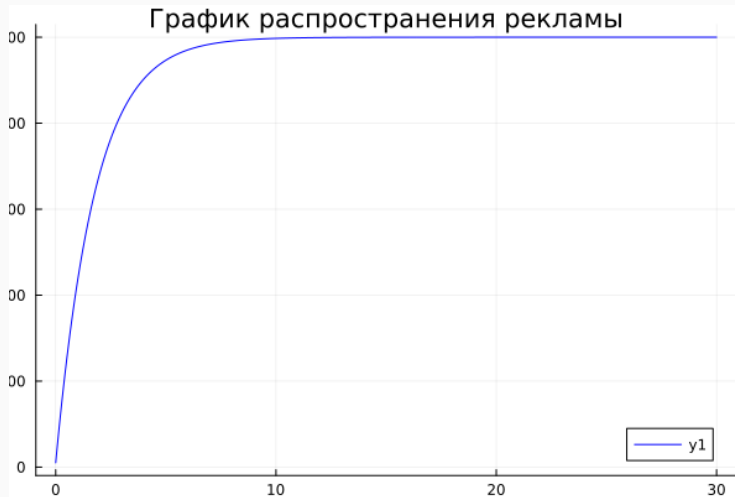
- Язык `Julia` и ее библиотеки: `Plots` и `DifferentialEquations` для построения графиков
- Свободное открытое программное обеспечение `OpenModelica` для моделирования ситуации

Ход работы

Напишем код программы на Julia

```
1jl
1 using Plots
2 using DifferentialEquations
3
4 N = 500
5 n0 = 5
6
7 function one(du, u, p, t)
8     (n) = u
9     du[1] = (0.55 + 0.0001*u[1])*(N - u[1])
10 end
11
12 v0 = [n0]
13 prom = (0.0, 30.0)
14 prob = ODEProblem(one, v0, prom)
15 sol = solve(prob, dtmax = 0.05)
16 A = [u[1] for u in sol.u]
17 T = [t for t in sol.t]
18
19 plt = plot(title = "График распространения рекламы", legend = true)
20 plot!(plt, T, A, color = :blue)
21
22 savefig(plt, "lab7_1.png")
```

В результате получаем следующий график



Напишем код для второй программы на Julia

```
2jl
1 using Plots
2 using DifferentialEquations
3
4 N = 500
5 n0 = 5
6
7 function two(du, u, p, t)
8     (n) = u
9     du[1] = (0.00005 + 0.2*u[1])*(N - u[1])
10 end
11
12 v0 = [n0]
13 prom = (0.0, 0.1)
14 prob = ODEProblem(two, v0, prom)
15 sol = solve(prob)
16 A = [u[1] for u in sol.u]
17 T = [t for t in sol.t]
18
19 mdn = 0;
20 mdnt = 0;
21 mdnn = 0;
22 for (i, t) in enumerate(T)
23     if sol(t, Val{1})[1] > mdn
24         global mdn = sol(t, Val{1})[1]
25         global mdnt = t
26         global mdnn = A[i]
27     end
28 end
29
30 plt = plot( dpi = 300, title = "График распространения рекламы", legend = tr
31 plot!( plt, T, A, color = :blue)
32 plot!(plt, [mdnt], [mdnn], seriestype = :scatter, color = :red)
33
34 savefig(plt, "lab7_2.png")
```

В результате получаем следующий график



Напишем код для третьей программы на Julia

```
3.jl
1  using Plots
2  using DifferentialEquations
3
4  N = 500
5  n0 = 5
6
7  function three(du, u, p, t)
8      (n) = u
9      du[1] = (0.3*sin(t) + 0.3*cos(t)*u[1])*(N - u[1])
10 end
11
12 v0 = [n0]
13 prom = (0.0, 0.1)
14 prob = ODEProblem(three, v0, prom)
15 sol = solve(prob, dtmax = 0.05)
16 A = [u[1] for u in sol.u]
17 T = [t for t in sol.t]
18
19 plt = plot( dpi = 300, title = "График распространения рекламы", legend = f
20 plot!( plt, T, A, color = :blue)
21
22 savefig(plt, "lab7_3.png")
```

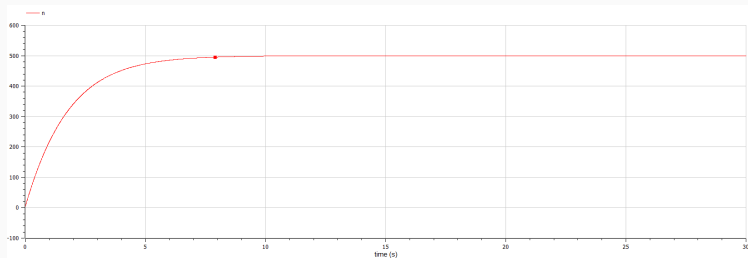
В результате получаем следующий график



Напишем код программы в OpenModelica

```
1 model one
2 Real N = 500;
3 Real n;
4 initial equation
5 n = 5;
6 equation
7 der(n) = (0.55 + 0.0001*n) * (N-n);
8 end one;
```

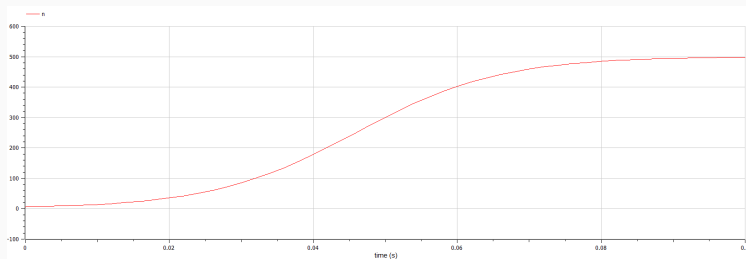
В результате получаем следующий график



Напишем код программы для второго случая в OpenModelica

```
1 model two
2 Real N = 500;
3 Real n;
4 initial equation
5 n = 5;
6 equation
7 der(n) = (0.00005 + 0.2*n) * (N-n);
8 end two;
```

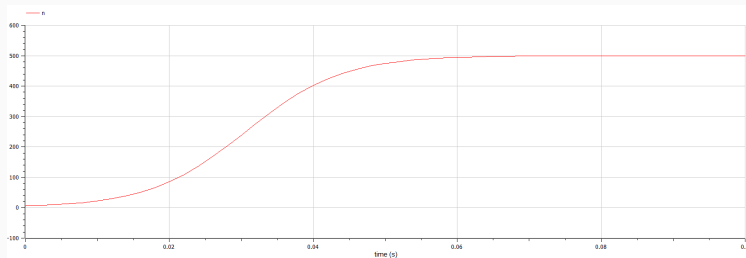
В результате получаем следующий график



Напишем код программы для третьего случая в OpenModelica

```
1 model three
2 Real N = 500;
3 Real n;
4 initial equation
5 n = 5;
6 equation
7 der(n) = (0.5*sin(time) + 0.3*cos(time)*n)*(N-n);
8 end three;
```

В результате получаем следующий график



Результаты

- Моделирование ситуации
- Ознакомление с языками
- Рассмотрение задачи об эффективности рекламы
- Построение графиков распространения рекламы, математическая модель которой описывается одним из трех уравнений
- Рассмотрение, как будет протекать эффективность рекламы в трех случаях