Отчёт по лабораторной работе №3

дисциплина: Математическое моделирование

Ерёменко Артём Геннадьевич, НПИбд-01-18

Содержание

# Цель работы

Построить упрощенную модель боевых действий с помощью Julia.

# Задание

**Вариант 4** Между страной и страной идет война. Численности состава войск исчисляются от начала войны и являются временными функциями и . В начальный момент времени страна имеет армию численностью 35 000 человек, а в распоряжении страны армия численностью в 49 000 человек. Для упрощения модели считаем, что коэффициенты постоянны. Также считаем и непрерывными функциями.

Постройте графики изменения численности войск армии и армии для следующих случаев:

1. Модель боевых действий между регулярными войсками
2. Модель ведение боевых действий с участием регулярных войск и партизанских отрядов

# Выполнение лабораторной работы

**1. Боевые действия между регулярными войсками**

1.1. Изучил начальные условия. Коэффициент смертности, не связанный с боевыми действиями, у первой армии 0,55, а у второй – 0,63. Коэффициент эффективности первой и второй армии 0,9 и 0,8 соответственно. Функция, описывающая подход подкрепление первой армии, , подкрепление второй армии описывается функцией . – численность 1-ой армии, – численность 2-ой армии.

1.2. Оформил начальные условия в код на Julia:

x0 = 35000 #численность первой армии  
y0 = 49000 #численность второй армии  
  
a = 0.55 #константа, характеризующая степень влияния различных факторов на потери  
b = 0.9 #эффективность боевых действий армии у  
c = 0.8 #эффективность боевых действий армии х  
h = 0.63 #константа, характеризующая степень влияния различных факторов на потери  
par = [a,b,c,h] #массив коэффициентов  
  
function P(t) #возможность подхода подкрепления к армии х  
 return (2\*abs(sin(t)))  
end  
  
function Q(t) #возможность подхода подкрепления к армии у  
 return (cos(13\*t) + 1)  
end

1.3. Для времени задал следующие условия: – начальный момент времени, – предельный момент времени.

1.4. Добавил в программу условия, описывающие время:

t0 = 0.0 #начальный момент времени  
tmax = 1.0 #предельный момент времени  
t = (t0,tmax)

1.5. Запрограммировал заданную систему дифференциальных уравнений, описывающих изменение численности армий:

#Система дифференциальных уравнений  
function NO\_PARTIZANS(du,u,p,t)  
 a,b,c,h = p  
 y1,y2 = u  
 du[1] = - a.\*y1 - b\*y2 + P(t) #изменение численности первой армии  
 du[2] = - c.\*y1 - h\*y2 + Q(t) #изменение численности второй армии  
end

1.6. Создал вектор начальной численности армий:

v0 = [x0,y0] #Вектор начальных условий

1.7. Запрограммировал решение системы уравнений:

prob = ODEProblem(NO\_PARTIZANS,v0,t,par)  
sol = solve(prob)

1.8. Описал построение графика изменения численности армий:

#Построение графиков решений  
plot(sol,xlabel = "Время", title = "Регулярные войска", label = ["Страна X" "Страна Y"])

**2. Боевые действия с участием регулярных войск и партизанских отрядов**

2.1. Изучил начальные условия. Коэффициент смертности, не связанный с боевыми действиями, у первой армии 0,35, а у второй – 0,6. Коэффициент эффективности первой и второй армии 0,46 и 0,2 соответственно. Функция, описывающая подход подкрепление первой армии, , подкрепление второй армии описывается функцией . Изначальная численность армий такая же, как и в п. 1.1.

2.2. Дополнил начальные условия в коде на Python:

a\_p = 0.35 #константа, характеризующая степень влияния различных факторов на потери  
b\_p = 0.46 #эффективность боевых действий армии у  
c\_p = 0.2 #эффективность боевых действий армии х  
h\_p = 0.6 #константа, характеризующая степень влияния различных факторов на потери  
par\_p = [a\_p,b\_p,c\_p,h\_p] #массив коэффициентов  
  
function P\_p(t) #возможность подхода подкрепления к армии х  
 return (1.5\*abs(sin(2\*t)))  
end  
  
function Q\_p(t) #возможность подхода подкрепления к армии у  
 return (cos(0.5\*t) + 1)  
end

2.3. Условия для времени оставил такие же, как и в п. 1.3, соответственно, не дублировал их в программе.

2.4. Запрограммировал заданную систему дифференциальных уравнений, описывающих изменение численности армий:

#Система дифференциальных уравнений  
function PARTIZANS(du,u,p,t)  
 a,b,c,h = p  
 y1,y2 = u  
 du[1] = - a\*y1 - b\*y2 + P\_p(t) #изменение численности первой армии  
 du[2] = - c\*y1\*y2 - h\*y2 + Q\_p(t) #изменение численности второй армии  
end

2.5. Т. к. начальная численность армий не изменилась, вектор начальных условий тоже не менял.

2.6. Запрограммировал решение системы уравнений:

#Решение системы  
prob\_p = ODEProblem(PARTIZANS,v0,(0.0,0.001),par\_p)  
sol\_p = solve(prob\_p)

2.7. Описал построение графика изменения численности армий:

#Построение графиков решений  
plot(sol\_p, title = "Регулярные войска + партизаны", label = ["Страна X" "Страна Y"])

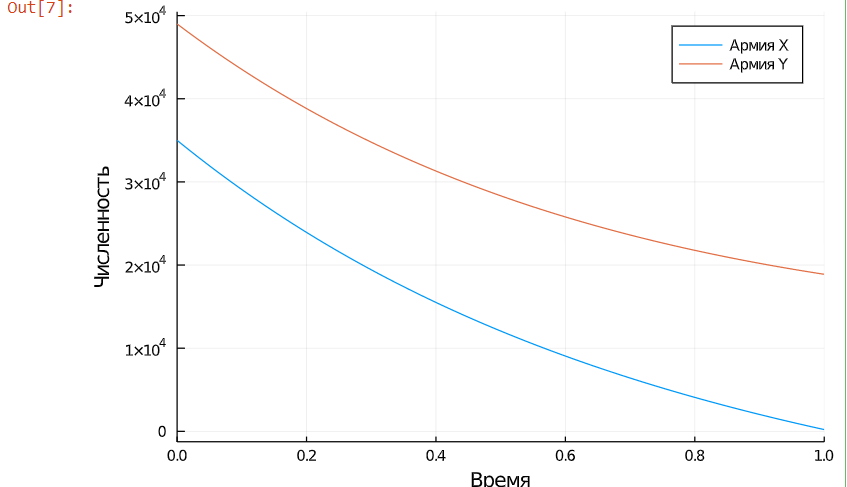
**3. Сборка программы**

3.1. Собрал код программы воедино и получила следующий код:

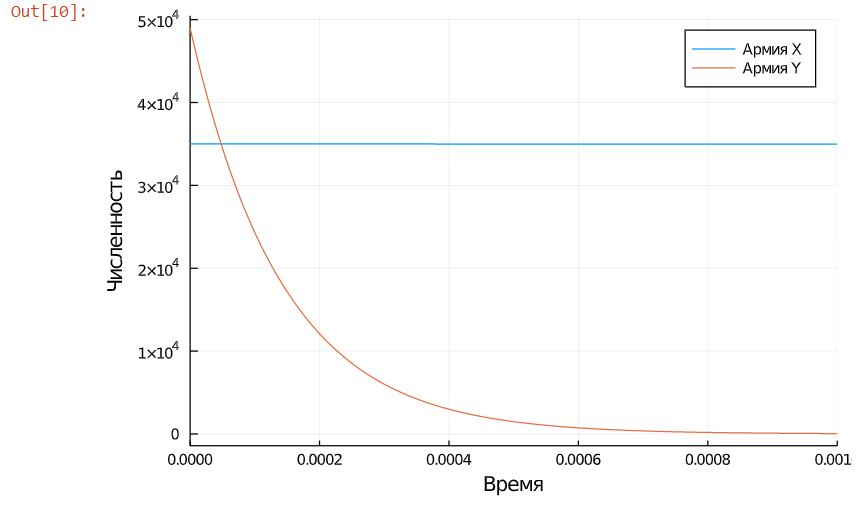
using DifferentialEquations, Plots  
  
x0 = 35000 #численность первой армии  
y0 = 49000 #численность второй армии  
  
a = 0.55 #константа, характеризующая степень влияния различных факторов на потери  
b = 0.9 #эффективность боевых действий армии у  
c = 0.8 #эффективность боевых действий армии х  
h = 0.63 #константа, характеризующая степень влияния различных факторов на потери  
par = [a,b,c,h] #массив коэффициентов  
  
function P(t) #возможность подхода подкрепления к армии х  
 return (2\*abs(sin(t)))  
end  
  
function Q(t) #возможность подхода подкрепления к армии у  
 return (cos(13\*t) + 1)  
end  
  
t0 = 0.0 #начальный момент времени  
tmax = 1.0 #предельный момент времени  
t = (t0,tmax)  
  
#Система дифференциальных уравнений  
function NO\_PARTIZANS(du,u,p,t)  
 a,b,c,h = p  
 y1,y2 = u  
 du[1] = - a.\*y1 - b\*y2 + P(t) #изменение численности первой армии  
 du[2] = - c.\*y1 - h\*y2 + Q(t) #изменение численности второй армии  
end  
  
v0 = [x0,y0] #Вектор начальных условий  
  
#Решение системы  
prob = ODEProblem(NO\_PARTIZANS,v0,t,par)  
sol = solve(prob)  
  
#Построение графиков решений  
plot(sol,xlabel = "Время", title = "Регулярные войска", label = ["Страна X" "Страна Y"])

a\_p = 0.35 #константа, характеризующая степень влияния различных факторов на потери  
b\_p = 0.46 #эффективность боевых действий армии у  
c\_p = 0.2 #эффективность боевых действий армии х  
h\_p = 0.6 #константа, характеризующая степень влияния различных факторов на потери  
par\_p = [a\_p,b\_p,c\_p,h\_p] #массив коэффициентов  
  
function P\_p(t) #возможность подхода подкрепления к армии х  
 return (1.5\*abs(sin(2\*t)))  
end  
  
function Q\_p(t) #возможность подхода подкрепления к армии у  
 return (cos(0.5\*t) + 1)  
end  
#Система дифференциальных уравнений  
function PARTIZANS(du,u,p,t)  
 a,b,c,h = p  
 y1,y2 = u  
 du[1] = - a\*y1 - b\*y2 + P\_p(t) #изменение численности первой армии  
 du[2] = - c\*y1\*y2 - h\*y2 + Q\_p(t) #изменение численности второй армии  
end  
  
#Решение системы  
prob\_p = ODEProblem(PARTIZANS,v0,(0.0,0.001),par\_p)  
sol\_p = solve(prob\_p)  
#Построение графиков решений  
plot(sol\_p, title = "Регулярные войска + партизаны", label = ["Страна X" "Страна Y"])

3.2. Получил графики изменения численностей армий (см. рис. @fig:001 и @fig:002):



Боевые действия между регулярными войсками



Боевые действия с участием регулярных войск и партизанских отрядов

# Выводы

Построил упрощенную модель боевых действий с помощью Julia.

В боевых действиях между регулярными войсками победит армия Y, причем ей на это потребуется довольно много времени (видим по графику, что численность армии X будет на исходе практический в предельный момент времени).

В боевых действиях с участием регулярных войск и партизанских отрядов уже победит армия Х, при чём довольно быстро (видим по графику, что армия Y потеряла всех бойцов практически сразу после начала войны).