

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное автономное образовательное учреждение высшего образования

**«Дальневосточный федеральный университет»**

**(ДВФУ)**

|  |
| --- |
| **ИНСТИТУТ МАТЕМАТИКИ И КОМПЬЮТЕРНЫХ ТЕХНОЛОГИЙ**  **Департамент математического и компьютерного моделирования** |

**Реферат**

по дисциплине

«Уравнения математической физики»

по теме:

«Математическая модель переноса вирусов в виде ОДУ»

направление подготовки 02.03.01 «Математика и компьютерные науки»

направление образовательной программы

«Сквозные цифровые технологии»

|  |  |
| --- | --- |
|  | Выполнил студент  гр. Б9119-02.03.01сцт  Петров С.Д.  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_ |
|  | *(Ф.И.О.) (подпись)*  Проверил  Алексеев Г. В.  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  *(Ф.И.О.) (подпись)*  «01»\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_2023г. |
|  | февраля |

г. Владивосток

2023

**Оглавление**

[1. Введение 3](#_Toc126182776)

[2. Модель “SIR” 3](#_Toc126182777)

[2.1. Введение 3](#_Toc126182778)

[2.2. Описание процесса 3](#_Toc126182779)

[2.3. Обозначение переменных 4](#_Toc126182780)

[2.4. Построение математической модели 4](#_Toc126182781)

[2.5. Программная реализация 5](#_Toc126182782)

[2.5.1. Код реализации 6](#_Toc126182783)

[2.5.2. Графики 7](#_Toc126182784)

[2.5.3. Анализ графиков 8](#_Toc126182785)

[2.6. Вывод 8](#_Toc126182786)

[3. Модель “WIRiv” 9](#_Toc126182787)

[3.1. Введение 9](#_Toc126182788)

[3.2. Описание процесса 9](#_Toc126182789)

[3.3. Обозначение переменных 9](#_Toc126182790)

[3.4. Построение математической модели 10](#_Toc126182791)

[3.5. Программная реализация 11](#_Toc126182792)

[3.5.1. Код реализации 13](#_Toc126182793)

[3.5.2. Графики численных экспериментов 16](#_Toc126182794)

[3.5.3. Анализ графиков 17](#_Toc126182795)

[3.6. Вывод 17](#_Toc126182796)

[4. Заключение 18](#_Toc126182797)

1. Введение

Эпидемии издавна угрожали человечеству, и только в ХХ веке были разработаны эффективные средства борьбы с инфекциями.

Проблема вирусной заболеваемости и возникновение эпидемий требует оперативного решения и активно исследуется во всем мире, в том числе – и с использованием математического аппарата и компьютеров. Математическое компьютерное моделирование используется для исследования механизмов распространения болезни. Такие модели позволяют прогнозировать и оценивать динамики передачи заболеваний. Это позволяет анализировать и контролировать ситуацию, связанную с распространением вируса, а также предугадывать серьёзные последствия и принимать соответствующие меры по их устранению.

1. Модель “SIR”
   1. Введение

Данная модель была разработана в 1927 году и показывает распространение вируса в популяции особей. В модели заболевание передается при контакте здоровой особи с зараженной.

* 1. Описание процесса

Пусть в момент времени 0 имеются:

Восприимчивые особи из множества инфицируются при контакте с особями из множества со скоростью .

Инфицированные особи получают иммунитет со скоростью .

* 1. Обозначение переменных

Обозначим переменные, необходимые для данной модели:

* *с* – скорость заражения
* *w* – скорость выздоровления
* *S(t)* – число особей восприимчивых к инфекции
* *I(t)* – количество инфицированных особей
* *R(t)* – количество выздоровевших особей
  1. Построение математической модели

Для начала определим, как меняется число особей, подверженных заражению. Оно может измениться от контакта с инфицированными.

**Комментарий:** *из этого уравнения видно, что число восприимчивых к инфекции особей уменьшается в зависимости от числа инфицированных, числа не инфицированных и скорости заражения. Таким образом, моделируется заражение особей при контакте.*

Скорость изменения количества особей с иммунитетом имеет следующий вид:

**Комментарий*:*** *отсюда видно – число выздоровевших особей меняется в зависимости от числа инфицированных особей и скорости выздоровления.*

Составим уравнение, выражающее скорость изменения количества инфицированных особей:

**Комментарий:** *можно заметить, что число инфицированных особей увеличивается на количество только что зараженных особей и одновременно уменьшается на число только что выздоровевших особей.*

* 1. Программная реализация

Для работы с данной моделью решать систему дифференциальных уравнения будем численно. Для этого возьмем метод Эйлера в связи с простотой его реализации. Необходимо найти значения в момент времени . Для этого будут рассчитываться . Высчитываются значения в следующий момент времени по формуле соответственно

* + 1. Код реализации

class SIR:

def \_\_init\_\_(self, S, I, R, c, w, h):

self.sum = S + I + R

self.S, self.I, self.R = S / self.sum, I / self.sum, R / self.sum

self.c, self.w = c / self.sum, w / self.sum

self.h = h

def create\_time\_line(self, max\_timer, step=1):

S\_line = []

I\_line = []

R\_line = []

S, I, R = self.S, self.I, self.R

for t in range(int(max\_timer / self.h) + 1):

if t % step == 0:

S\_line += [S]

I\_line += [I]

R\_line += [R]

S, I, R = (

S - self.c \* self.h \* I \* S,

I + self.c \* self.h \* I \* S - self.w \* self.h \* I,

R + self.w \* self.h \* I,

)

return (

np.array(S\_line) \* self.sum,

np.array(I\_line) \* self.sum,

np.array(R\_line) \* self.sum,

)

def create\_plot(self, max\_timer, step):

S\_list, I\_list, R\_list = self.create\_time\_line(max\_timer, step)

steps = S\_list.shape[0]

x = range(steps)

plt.fill\_between(x, [0] \* steps, S\_list, label="Подверженные заражению")

plt.fill\_between(x, S\_list, I\_list + S\_list, label="Зараженные")

plt.fill\_between(

x, S\_list + I\_list, S\_list + I\_list + R\_list, label="Выздоровевшие"

)

plt.xlabel("Часы")

plt.ylabel("Особи")

plt.legend()

plt.show()

* + 1. Графики

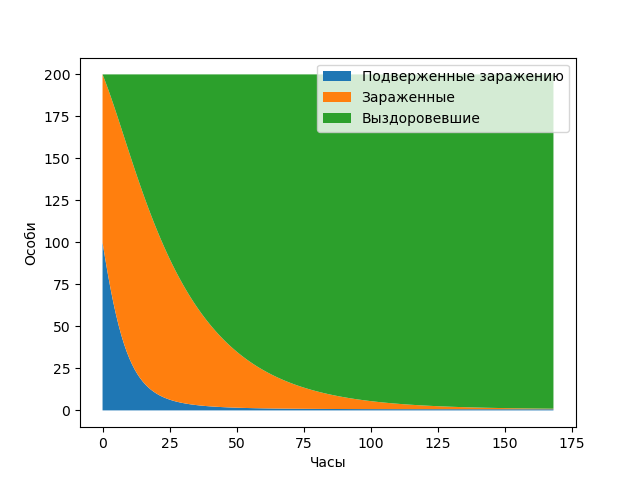


Рис. 1

Параметры:

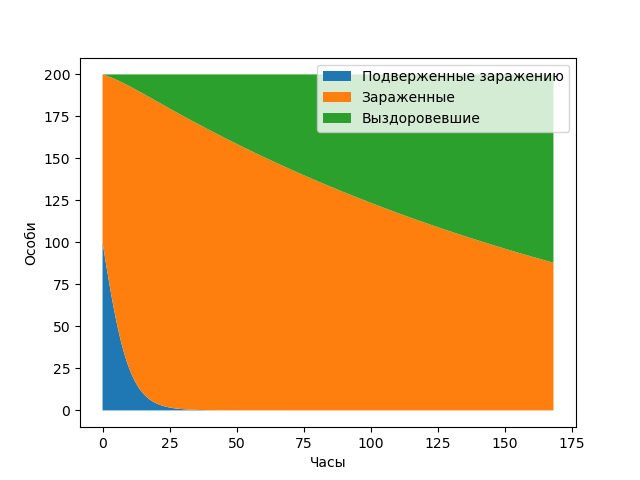


Рис. 2

Параметры:

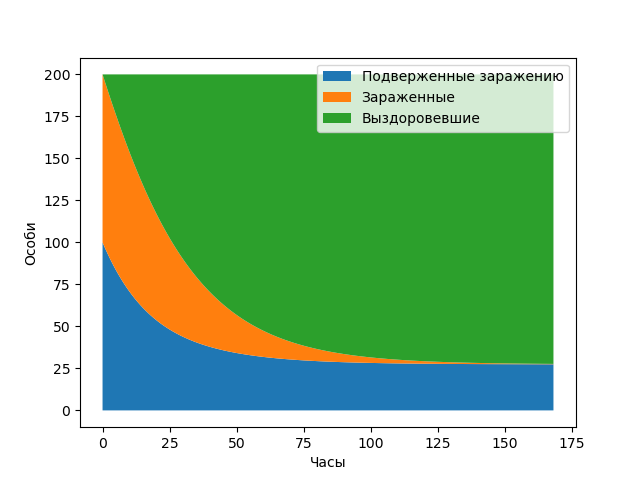


Рис. 3

Параметры:

* + 1. Анализ графиков

На всех графиках отображен период в неделю с начала численного эксперимента. Количество зараженных и незараженных особей равны 100. Шаг по времени для моделирования – 1 минута.

На Рис. 1 изображен случай, в котором из-за скорости заражения и выздоровления вся эпидемия заканчивается через 5 дней и не остается не переболевших особей.

На Рис. 2 - из-за медленной скорости выздоровления эпидемия продолжается слишком долго и ее завершение не влезает в смоделированный промежуток.

На Рис. 3 – из-за достаточно большой скорости выздоровления эпидемия заканчивается до того момента, когда все особи переболели.

* 1. Вывод

Данная модель учитывает основные отношения между изменениями количества особей в разных группах (S, I, R). Однако модель не учитывает распределение особей по территории, способ передачи заболевания, а также период, после которого особь не восприимчива к вирусу.

1. Модель “WIRiv”
   1. Введение

Данная модель описывает процессы, происходящие внутри организма, а именно распространение вируса внутри организма на клеточном уровне.

* 1. Описание процесса

Пусть

Клетки, которые были в организме изначально(), под воздействием вирионов или молекул интерферона могут стать зараженными или иммунными соответственно. Через определенный промежуток времени определенная доля интерферонов, вирионов и инфицированных клеток удаляется. Также инфицированные клетки производят вирионы, и на каждую инфицированную клетку организм производит интерферон.

* 1. Обозначение переменных

Обозначим переменные, необходимые для данной модели:

* *W(t)* – количество клеток подверженных инфицированию
* *I(t)* – количество инфицированных клеток
* *R(t)* – количество клеток, имеющих иммунитет
* *i(t)* – количество молекул интерферона
* *v(t)* – количество вирионов
* – количество интерферонов и вирионов, вырабатываемых на инфицированную клетку за единицу времени
* – доля удаляемого интерферона, вириона, инфицированных клеток за единицу времени
* – доля первозданных клеток, вступивших в контакт с интерфероном или вирионом в единицу времени
* – доля интерферонов или вирионов, которые уходят на взаимодействие клетками подверженными заражению
  1. Построение математической модели

Запишем уравнение изменения числа первозданных клеток:

Здесь:

Уравнение, отображающее скорость изменения зараженных особей:

Здесь:

Уравнение скорости изменения количества резистентных клеток:

Здесь:

Уравнение, выражающее скорость изменения числа вирионов:

Здесь:

Уравнение, отображающее скорость изменения числа молекул интерферона:

Здесь:

**Комментарий:** *заметим, что в последних двух уравнениях производная может иметь как отрицательный, так и положительный знак. Что значит, число вирионов и интерферонов может увеличиваться или уменьшаться.*

* 1. Программная реализация

Для работы с данной моделью решать систему дифференциальных уравнения будем численно. Для этого возьмем метод Эйлера в связи с простотой его реализации. Необходимо найти значения в момент времени . Для этого будут рассчитываться . Высчитываются значения в следующий момент времени по формуле соответственно

* + 1. Код реализации

import numpy as np

import matplotlib.pyplot as plt

class WIRiv:

def \_\_init\_\_(self, W, I, R, i, v, g, mu, alpha, beta, h, max\_interferon=1000):

self.ceil\_sum = W + I + R

self.W = W / self.ceil\_sum

self.I = I / self.ceil\_sum

self.R = R / self.ceil\_sum

self.molec = max\_interferon

self.i = i / self.molec

self.v = v / self.molec

self.g\_i = g["i"]

self.g\_v = g["v"]

self.m\_i = mu["i"]

self.m\_I = mu["I"]

self.m\_v = mu["v"]

self.a\_i = alpha["i"]

self.a\_v = alpha["v"]

self.b\_i = beta["i"]

self.b\_v = beta["v"]

self.h = h

def create\_time\_line(self, max\_timer, step=1):

W\_line = []

I\_line = []

R\_line = []

i\_line = []

v\_line = []

W = self.W

I = self.I

R = self.R

i = self.i

v = self.v

for t in range(int(max\_timer / self.h) + 1):

if t % step == 0:

W\_line += [W]

I\_line += [I]

R\_line += [R]

i\_line += [i]

v\_line += [v]

W, I, R, i, v = (

W - self.a\_i \* i \* W - self.a\_v \* v \* W,

I + self.a\_v \* v \* W - self.m\_I \* I,

R + self.a\_i \* i \* W,

i + self.g\_i \* I - self.b\_i \* i \* W - self.m\_i \* i,

v + self.g\_v \* I - self.b\_v \* v \* W - self.m\_v \* v,

)

return (

np.array(W\_line) \* self.ceil\_sum,

np.array(I\_line) \* self.ceil\_sum,

np.array(R\_line) \* self.ceil\_sum,

np.array(i\_line) \* self.molec,

np.array(v\_line) \* self.molec,

)

def create\_plot(self, max\_timer, step):

W\_line, I\_line, R\_line, i\_line, v\_line = self.create\_time\_line(max\_timer, step)

steps = W\_line.shape[0]

x = range(steps)

fig, (ax1, ax2) = plt.subplots(1, 2, figsize=(16, 6))

ax1.fill\_between(x, [0] \* steps, W\_line, label="Подверженные заражению")

ax1.fill\_between(x, W\_line, I\_line + W\_line, label="Зараженные")

ax1.fill\_between(

x, W\_line + I\_line, W\_line + I\_line + R\_line, label="Выздоровевшие"

)

ax1.legend()

ax2.plot(x, i\_line, label="Интерферон")

ax2.plot(x, v\_line, label="Вирион")

ax2.legend()

plt.show()

* + 1. Графики численных экспериментов

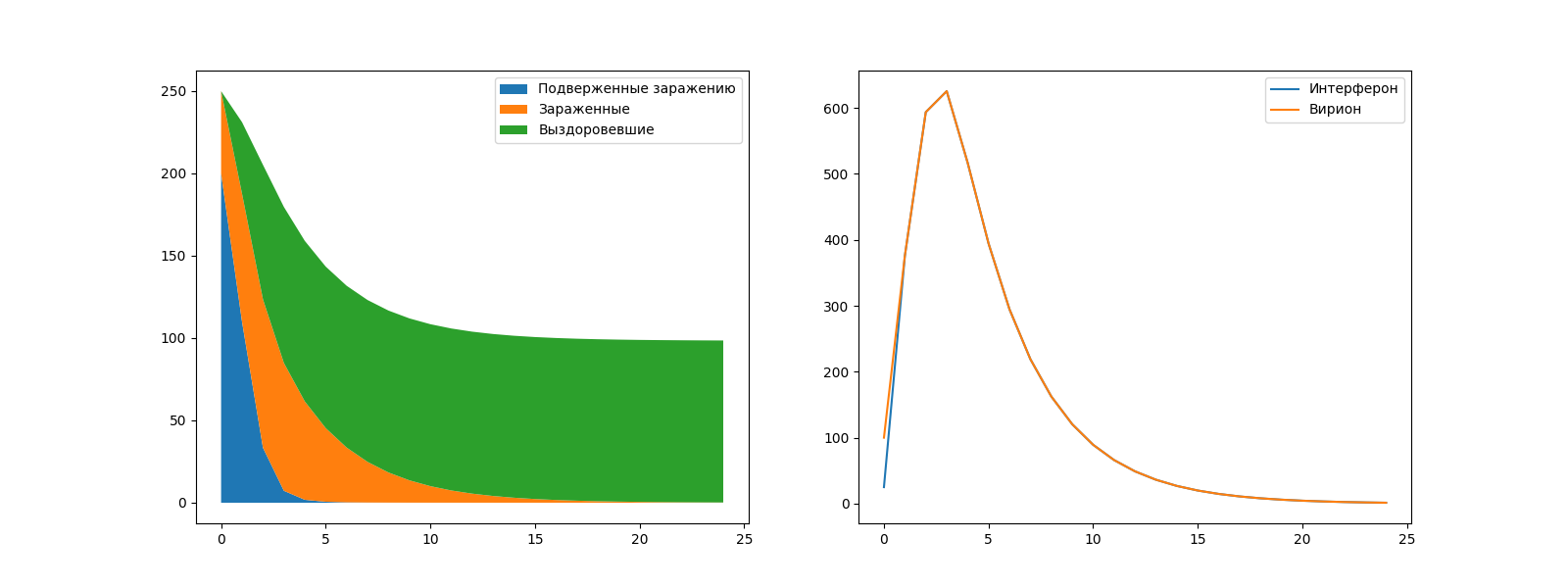


Рис. 4

Параметры:

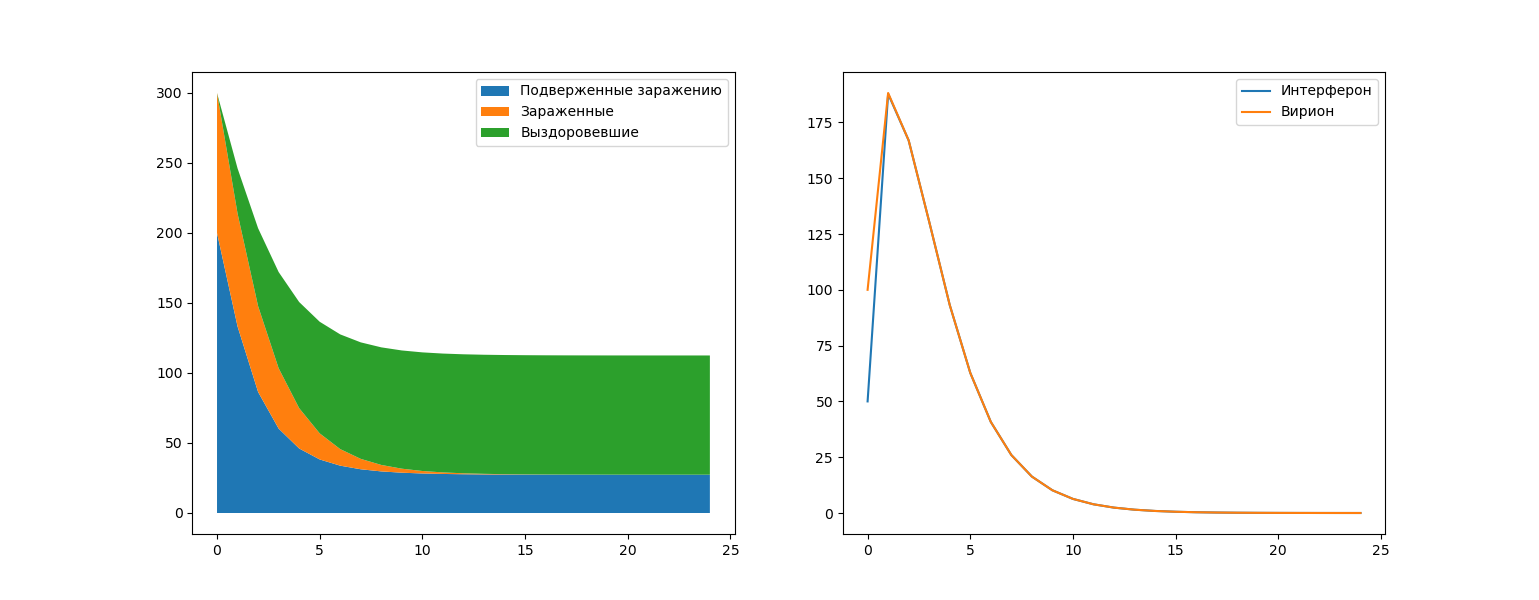


Рис. 5

Параметры:

* + 1. Анализ графиков

На графиках отображен промежуток в 24 часа с начала численного эксперисмента.

На Рис. 4 параметры подобранны таким образом, что за первые 6 часов все клетки заражаются и далее медленно вымирают. Удачно, что при увеличении числа зараженных клеток увеличивается число интерферона, а следственно, клетки получают иммунитет гораздо быстрее.

На Рис. 5 зараженные клетки слишком быстро вымирают, вследствие чего не все первозданные клетки получают иммунитет.

* 1. Вывод

Данная модель учитывает наличие интерферона и вирионов, а также скорости их выведения.

1. Заключение

В данной работе были рассмотрены математические модели по распространению вирусов и инфекций, описанные при помощи ОДУ. Были созданы реализации, помогающие познакомиться с результатами работы данных моделей.