MATH 442 HIV Project

Gage Broberg, Adrian Martinez, Kyle Moore

April 8 2022

1 Introduction

Systems of differential equations are frequently used to model natural phenomena such as the relationship between predators and prey, the position of a mass attached to a spring, epidemics, etc. Differential systems of equations are often easier to work with than regular systems of equations because they are less complex, are easier to formulate since each addition made represents some physical quality of the modeled elements, and are therefore more intuitive.

We initially focus on a pair of systems of differential equations used to model the interaction of the HIV-1 virus in the body. These systems were first proposed in a scientific article written by Alan S. Perelson, Avidan U. Neumann, Martin Markowitz, John M. Leonard and David D. Ho and published in a magazine called "Science" on March 15, 1996. We begin our analysis by detailing what real-world process each of the terms in the systems represents. We then use the two systems to replicate the derivation of equation (6) from the article. Using the values in the table at the bottom of page 1583 of the article, we also replicate the authors' determination of virion production and clearance rate for patient 107 which is: $V = 120.2113$, $c = 5.1725$, and $\delta = 0.4477$. Then we use these results to determine the number of virions produced per day in the entire body over the course of the entire HIV infection as well as (1) the average life-span of a virion; (2) the average life-span of an infected T-cell (productively infected T-cell); and (3) the average viral generation time, which are:

number of virions produced per day in the entire body over the course of the entire HIV infection $= 8.1 \times 10^9$

(1) the average life-span of a virion = 0.1933 days

(2) the average life-span of an infected T-cell (productively infected T-cell) = 2.2338 days

(3) the average viral generation time = 2.4271 days

Finally, we again use our solution for $V(t)$ and the information in the table to find an equilibrium point, for which the value comes out to: (412, 120.2113, 138.89)

Next, we attempt to use the perspective gained from the first part of this project to develop our own model for HIV-1 that captures the first two phases of infection (prior to treatment), using equations (1) and (2) in the article as a starting place and specifying units of each parameter. We then use our equation for $V(t)$ along with the equilibrium point previously derived to obtain values for the parameters of our model. After determining the equilibrium points in our new model, we conduct an analysis of the stability of each point. Using provided initial values, we provide a stacked plot of uninfected T cells as a function of time on top and virions as a function of time on bottom in order to verify our results. We determine the total number of T cells created per day during phase II of the infection. Next, we attempt to incorporate the final phase (phase III) of the HIV-1 infection by decreasing the production of T cells during phase II. Finally, we combine the model in the article represented by equations (3)-(4)-(5) and our previous model that included all three phases in order to model the effect of ritonavir on the infection.

2 Introduction of the Original Models

Let us first start by defining the variables used in the paper:

$$
\begin{aligned}
k &= \text{rate constant for target cells} \\
c &= \text{rate constant for virion clearance} \\
V &= \text{concentration of viral particles in plasma} \\
T &= \text{target cells} \\
T^* &= \text{productively infected cells} \\
\delta &= \text{rate of loss of virus producing cells} \\
N &= \text{number of new virions produced per infected cell over its lifetime} \\
V_I &= \text{plasma concentration of virions in the infectious pool} \\
V_{NI} &= \text{plasma concentration of virions in the non-infectious pool}
\end{aligned}
$$

The first system represents the change in populations of target cells and concentration of viral particles in the plasma in the body, assuming no treatment has been received:

$$\frac{dT^*}{dt} = kVT - \delta T^* \tag{1}$$

$$\frac{dV}{dt} = N\delta T^* - cV \tag{2}$$

In equation (1), the first term represents the interaction between target cells and viral particles and the resultant increase in productively infected cells. The second term represents the loss of infected cells which could be the result of viral cytopathicity, immune elimination, or other processes such as apoptosis according to the article. In equation (2), the first term represents the increase in viral particles in plasma. The second term represents the clearance of viral particles.

The second system represents the change in populations of target cells, concentration of infectious viral particles in the plasma, and concentration of non-infectious viral particles in the plasma, assuming treatment with an antiviral called ritonavir:

$$\frac{dT^*}{dt} = kV_IT - \delta T^* \tag{3}$$

$$\frac{dV_I}{dt} = -cV_I \tag{4}$$

$$\frac{dV_{NI}}{dt} = N\delta T^* - cV_{NI} \tag{5}$$

Equation (3) is effectively the same as equation (1), except that only target cell interaction with infectious viral particles leads to an increase in the number of infected target cells. Equation (4) represents the clearance of viral particles from the population of infected. Equation (5) is effectively the same as equation (2), except that all new viral particles produced are non-infectious since the viral inhibition by ritonavir is assumed to be 100%.

3 Derivation of equation (6)

Equations (1) and (2)

We begin by evaluating equations (1) and (2). Before therapy has begun, viral loads are constant. Therefore we can set

$$dV/dt = 0$$

which, from equation (2), implies that

$$N\delta T_0^* = cV_0 \tag{6}$$

and since V is constant for weeks before therapy, T^* must also be constant, so

$$dT^*/dt = 0$$

which, from equation (1), implies that

$$kV_0T_0 = \delta T_0^* \tag{7}$$

Solving for T_0^* in equation (7), we get that

$$T_0^* = \frac{kV_0T_0}{\delta}$$

and plugging this value for T_0^* into equation (6), we get that:

$$N\cancel{\delta}\frac{k\cancel{V_0}T_0}{\cancel{\delta}} = c\cancel{V_0} \implies c = NkT_0 \tag{8}$$

so that c is in a quasi-steady state. If we assume on a scale of weeks that the T-cell count as well as V_0 and T_0 do not change then we can derive the following equations:

$$T_0 = \frac{c}{Nk} \tag{9}$$

$$T_0^* = \frac{cV_0}{N\delta} \tag{10}$$

$$V_0 = \frac{\delta}{kT_0}T_0^* \tag{11}$$

$$V_0 = \frac{N\delta}{c}T_0^* \tag{12}$$

So we get the general solution for the equations below:

$$T^*(t) = T_0^* e^{-\delta t} \tag{13}$$

and

$$V(t) = V_0e^{-ct} + \frac{N\delta T_0^*}{c-\delta}(e^{-\delta t} - e^{-ct}) \tag{14}$$

and assuming quasi-steady state before treatment, we use equation (8) and

$$T_0^* = \frac{kV_0T_0}{\delta} \tag{15}$$

to get that

$$V(t) = \frac{V_0}{c-\delta}(ce^{-\delta t} - \delta e^{-ct}) \tag{16}$$

Equations (3) through (5)
Now using equations (3)-(5) we know that before therapy is initiated, $V_{NI}(0) = 0$ and all of the virus belongs to the infectious pool, which means

$$V_I(0) = V_0 \implies V_I(t) = V_0e^{-ct} \tag{17}$$

After a short period of time, we can assume that $T = T_0$ so we can solve (3) for

$$T^*(t) = T^*(0)e^{-\delta t} + \frac{kV_0T_0}{\delta(c-\delta)}[e^{-\delta t} - e^{-ct}] \tag{18}$$

and assuming that T^* is quasi-steady state before infection,

$$T^*(0) = \frac{kV_0T_0}{\delta} \tag{19}$$

and thus

$$T^*(t) = \frac{kV_0T_0}{\delta(c-\delta)}(ce^{-\delta t} - \delta e^{-ct}) \tag{20}$$

and by observing that $T^*(t) = T_0^*$, we can simplify $T^*(t)$ to

$$T^*(t) = \frac{T_0^*}{c-\delta}(ce^{\delta t} - \delta e^{-ct}) \tag{21}$$

Plugging this into the ODE for V_{NI} helps us obtain that

$$V_{NI}(t) = \frac{cV_0}{c-\delta}(\frac{c}{c-\delta}(e^{-\delta t} - e^{-ct}) - \delta te^{-ct}) \tag{22}$$

and since $V = V_I + V_{NI}$, we can obtain the desired equation for V(t):

$$V_{NI}(t) = V_0e^{-ct} + \frac{cV_0}{c-\delta}(\frac{c}{c-\delta}(e^{-\delta t} - e^{-ct}) - \delta te^{-ct}) \tag{23}$$

MATLAB

We can then use MATLAB to programmatically confirm that equation (6) is the solution for $V(t)$. We obtain the following result from the script file (Note that the MATLAB script file used is included in the Appendix):

$$T^*(t) = \frac{T_0^*}{c-\delta}[ce^{-\delta t} - \delta e^{-ct}] \tag{24}$$

$$V_{NI}(t) = \frac{cV_0}{c-\delta}[\frac{c}{c-\delta}(e^{-\delta t} - e^{-ct}) - \delta t e^{-ct}] \tag{25}$$

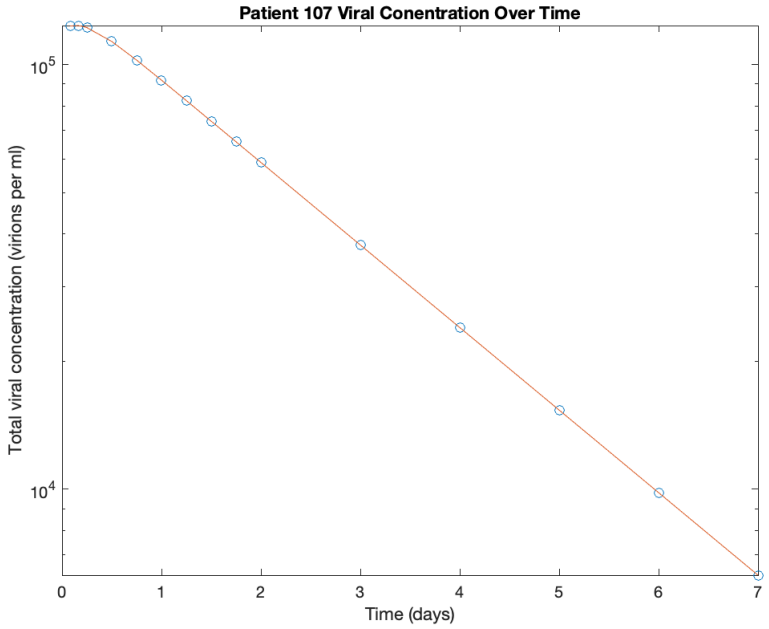
$$V_I(t) = V_0 e^{-ct} \tag{26}$$

and since $V(t) = V_I(t) + V_{NI}(t)$, we confirm that indeed,

$$V(t) = V_I(t) + V_{NI}(t) = V_0 e^{-ct} + \frac{cV_0}{c-\delta}[\frac{c}{c-\delta}(e^{-\delta t} - e^{-ct}) - \delta t e^{-ct}] \tag{27}$$

4 Fitting Patient 107 Data

We now have a basic model for $V(t)$. Next, we attempt to improve this model by fitting it to the data of patient 107 using the MATLAB function lsqcurvefit and we obtain the following plot of viral concentration versus time for patient 107 (Note that the MATLAB script file used is included in the Appendix):



and the following parameter values:

$$V_0 = 120.2113$$
$$c = 5.1725$$
$$\delta = 0.4477$$

Finding the number of virions produced per day

According to the 3rd column of page 1583 of the article, at steady state, the production rate of virus must equal its clearance rate, cV_0.

$$cV_0 = 5.1275 \cdot 120.2113 = 621.79 \frac{\text{virions}}{mm^3 \cdot \text{day}} \tag{28}$$

According to the table caption, the total virion production in the article was calculated using the volume of extracellular fluid in the body. We can derive the amount of ECF used by the authors of the article using the following calculation:

$$\frac{3.1 * 10^9}{(77000 * 3.09)} = 13029.04 \tag{29}$$

Therefore, the total number of virions produced in the body is equal to:

$$621.79 \frac{\text{virions}}{\cancel{mm^3} \cdot \text{day}} * 1000 \frac{\cancel{mm^3}}{\cancel{\text{ml}}} * 13029.04 \frac{\cancel{\text{ml}}}{\text{body}} = 8.1 \times 10^9 \frac{\text{virions}}{\text{body} \cdot \text{day}} \tag{30}$$

We are then able to determine the equations for (1) the average life-span of a virion, (2) average life-span of an infected T-cell (productively infected T-cell) and (3) average viral generation time as follows:

$$(1)\ \frac{1}{c} = \frac{1}{5.1725} = 0.1933$$
$$(2)\ \frac{1}{\delta} = \frac{1}{0.4477} = 2.2338$$
$$(3)\ \frac{1}{c} + \frac{1}{\delta} = \frac{1}{5.1725} + \frac{1}{0.4477} = 0.1933 + 2.2338 = 2.4271$$

5 Finding the Equilibrium Point

To find the equilibrium point, we need the values T_e, V_e and T_e^*. We can take $T_e = 412$ to be the baseline number of T cells for patient 107. We can also use the derived value $V_e = 120.2113$ from section 3 for V_e. Our previously derived formula for T_e^* in section 3 was:

$$T_e^* = \frac{cV_e}{N\delta} \tag{31}$$

taking $N = 10$ and using the values of V_e, c, and δ derived from our non-linear fit in section 3, we get that:

$$T_e^* = \frac{5.1725 \cdot 120.2113}{10 \cdot 0.4477} = 138.89 \frac{10^3 \cdot \text{virions}}{mm^3} \tag{32}$$

So our equilibrium point is (412, 120.2113, 138.89).

6 Creating Our Own Model

Up to this point, we have solely been manipulating the models of the authors of the article. Now, we will attempt to create our own model. We begin with equation (1) and (2) and add a third equation to model the change of T with time. Our model is:

$$\frac{dT^*}{dt} = kVT - \delta T^* \tag{33}$$
$$\frac{dV}{dt} = N\delta T^* - cV \tag{34}$$
$$\frac{dT}{dt} = \beta T - kVT \tag{35}$$

where β is the rate of T-cell creation.

Units for parameters in the above model

$$k = \frac{mm^6}{\text{second}}$$
$$c = \frac{mm^3}{\text{second}}$$
$$\delta = \frac{mm^3}{\text{second}}$$
$$\beta = \frac{mm^3}{\text{second}}$$
$$N = \frac{1}{\text{T-cell}}$$

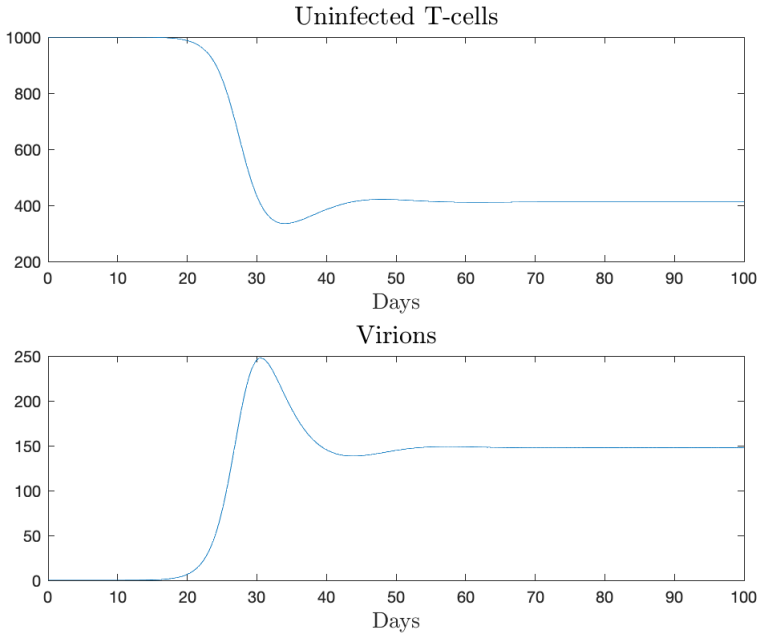
Furthermore, we can use MATLAB to find values values for the parameters of our model. The MATLAB script used to do this is included below in the appendix and titled modelParamters.m.

7 Stability

Using MATLAB, we were able to find the stability of both equilibrium points. The MATLAB code used is in the appendix below. We concluded that the equilibrium point $(1000, 0, 0$ is unstable since it has eigenvalues of the Jacobian matrix with positive real parts, while the equilibrium point $(412, 120.2113, 138.89)$ is stable since all eigenvalues of the Jacobian matrix have negative real parts.

8 Stacked Plot of T-cells and Virions

A stacked plot of T-cells versus time and virions versus time can be created with MATLAB. The MATLAB script used to do this is included below in the appendix and titled stackedPlot.m. The plots generated are:



9 Determination of Number of T-cells Created During Phase II

The number of T-cells created per day per mm^3 of extracellular fluid is given by:

$$\beta T = 0.13 * 412 \tag{36}$$

$$= 53.56 \text{ T-cells created per day per } mm^3 \tag{37}$$

According to figure 3 in the assignment, the average time for a patient to remain in phase II of the HIV infection is around 10 years. Using this information and the number of T-cells created per day, the number of T-cells created over an average HIV infection is given by:

$$53.56 * 10 \text{ years} * 365 \text{ days in a year} = 195494 \text{ T-cells created in phase II per } mm^3 \text{ of ECF} \tag{38}$$

so to convert this to total number of T-cells created in the body over phase II of HIV, we take a similar approach to the one we took when finding the number of virions produced per day. We previously found that the body of patient 107 contained 13029.04 ml of ECF. Therefore, it contains $13029.04 * 1000 = 13029042.16\, mm^3$ of ECF. So the total number of T-cells created in phase II comes out to:

$$1954494 * 13029042.16 = 2.55 * 10^{12} \tag{39}$$

T-cells.

10 Model w/ Ritonavir

Our model with ritonavir is as follows:

$$\frac{dT^*}{dt} = kV_I T - \delta T^* \tag{40}$$

$$\frac{dV_I}{dt} = -cV_I \tag{41}$$

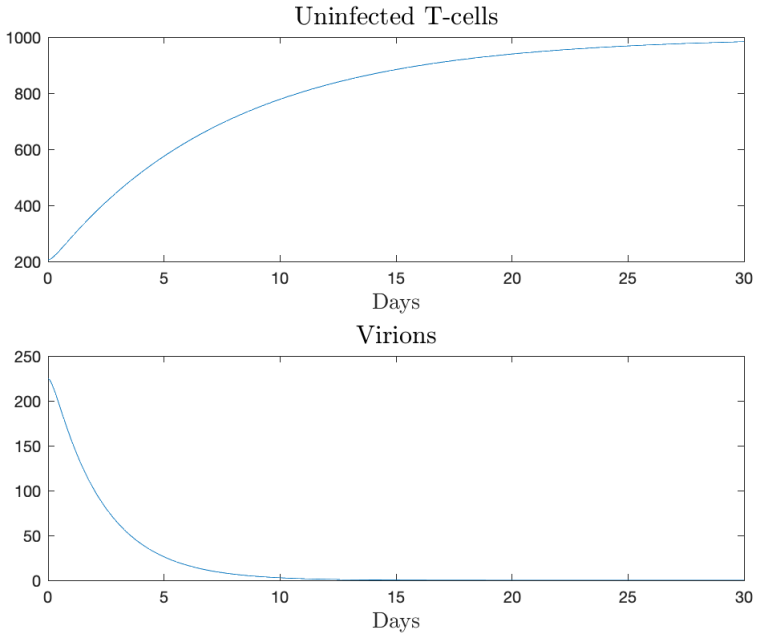
$$\frac{dV_{NI}}{dt} = N\delta T^* - cV_{NI} \tag{42}$$

$$\frac{dT}{dt} = \beta(1000 - T) - kVT \tag{43}$$

We then determine the point at which the patient has AIDS. This is approximately

$$T = 205 \tag{44}$$
$$T^* = = 225 \tag{45}$$
$$V_I = 350 \tag{46}$$
$$V_{NI} = 0 \tag{47}$$

and we obtain the following plots using the MATLAB scripts in the appendix titled stackedPlot13.m. The plots are:



and we can simply use the function in our MATLAB script to find the number of days after ritonavir administration at which the T-cell population returns to $900mm^{-3}$. This is 16.1543 days.

11 Summary

In conclusion, after studying the model of HIV infection proposed in 1996 we were able to produce our own model to describe the full progression HIV infection. In order to derive the equations provided, we had to use the values in the table at the bottom of page 1583 of the article. We also replicated the authors' determination of virion production and clearance rate for patient 107 for which the values were: $V = 120.2113$, $c = 5.1725$, and $\delta = 0.4477$. With this, we were then able to calculate the total number of virions produced in the body to be 8.1×10^9. We then created our own model where we were able to find the stability of both equilibrium points. Using MATLAB we were able to provide stacked plots of T-cells versus time and virions versus time. To determine the number of T-cells created during Phase II, we calculated the number of T-cells created per day and the number of T-cells created over an average HIV infection and got the number 195494. We previously found that the body of patient 107 contained 13029.04 ml of ECF. Therefore, we concluded that it contained $13029.04 * 1000 = 13029042.16\ mm^3$ of ECF. So the total number of T-cells created in phase II comes out to 2.55×10^{12} T-cells. Finally, we computed our model with ritonavir and we were able to obtain the plots using MATLAB to find the number of days after ritonavir administration where the T-cells population returns to $900mm^{-3}$ which came out to be 16.1543 days.

12 Appendix

DeriveEq6.m

```
syms Tstar(t) V1(t) VN(t) k T delta c N
eq3 = diff(Tstar) == (k*V1*T) - (delta*Tstar)
eq4 = diff(V1) == -c*V1
eq5 = diff(VN) == (N*delta*Tstar) - (c*VN)
eqs2 = [eq3 eq4 eq5]
[Tstar, V1, VN] = dsolve(eqs2)
```

DeriveEq6Output

```
DeriveEq6

eq3(t) =

diff(Tstar(t), t) == T*k*V1(t) - delta*Tstar(t)


eq4(t) =

diff(V1(t), t) == -c*V1(t)


eq5(t) =

diff(VN(t), t) == N*delta*Tstar(t) - c*VN(t)


eqs2(t) =

[diff(Tstar(t), t) == T*k*V1(t) - delta*Tstar(t), diff(V1(t), t) == -c*V1(t), diff(VN(t), t) == N*delta


Tstar =

(C3*T*k*exp(-delta*t))/(c - delta) - (C1*T*k*exp(-c*t))/(c - delta)


V1 =

C1*exp(-c*t)


VN =

(C3*N*T*delta*k*exp(-delta*t))/(c^2 - 2*c*delta + delta^2) - C1*((N*T*delta*k*exp(-c*t))/(c^2 - 2*c*del

diary off
```

Patient107model.m

```
% Question #4 V0 = p(1), c = p(2), delta = p(3)
times = [1/12 1/6 1/4 1/2 3/4 1 5/4 3/2 7/4 2 3 4 5 6 7];
viral = [119 137 111 119 89 103 89 58 67 72 30 25 16 8 6].*1000;
v = @(p,t) p(1)*exp(-p(2).*t)+(p(2)*p(1)/(p(2)-p(3)))*((p(2)/(p(2)-p(3)))*(exp(-p(3).*t)-exp(-p(2).*t))
% Central Differences
p0 = [77000 3.09 0.50];
[p ssr] = lsqcurvefit(v,p0,times,viral)
v0=p(1);
c=p(2);
delta=p(3);
%t=0:8;
modelviral=v(p,times)
semilogy(times,modelviral,'o', times, modelviral)
title("Patient 107 Viral Conentration Over Time")
xlabel("Time (days)");
ylabel("Total viral concentration (virions per ml)");
```

Patient107modelOutput

```
Patient107model

<a href = "matlab: helpview('optim','local_min_possible_lsq','CSHelpWindow');">Local minimum possible</

lsqcurvefit stopped because the final change in the sum of squares relative to
its initial value is less than the value of the <a href = "matlab: helpview('optim','function_tolerance

<<a href = "matlab: createExitMsg({'optimlib:snls:Exit3basic','lsqcurvefit'},{'optimlib:snls:Exit3detai

p =

   1.0e+05 *

    1.2021    0.0001    0.0000


ssr =

   1.1852e+09


modelviral =

   1.0e+05 *

  Columns 1 through 13

    1.2329    1.2364    1.2227    1.1338    1.0249    0.9194    0.8229    0.7360    0.6581    0.5885

  Columns 14 through 15

    0.0982    0.0628
```

stability.m

```
%initialize variables
syms t tstar v;
N = 10;
c = 5.1725;
B = 0.1305;
k = c/(N * 412);
delta = 0.4477;
v0 = 120.2113;

% system of equations
y = [tstar, v, t];
f=[k*v*t-delta*tstar, n*delta*tstar-c*v, B*t-k*v*t];

% jacobian matrix
J = jacobian(f,y);

% equilibrium point (1000, 0, 0)
t = 1000;
tstar = 0;
v = 0;
A2 = subs(J)
lamba = double(eig(A2))

% equilibrium point (412, 120.2113, 138.89)
t = 412;
v = 120.2113;
tstar = 138.89;
A1 = subs(J)
lamba = double(eig(A1))
```

stabilityOutput

```
stabiliy

A2 =

[-4477/10000,  2827047217545375/2251799813685248,        0]
[  4477/1000,                          -2069/400,        0]
[          0, -2827047217545375/2251799813685248, 261/2000]


lamba =

    0.1305
   -6.1570
    0.5368


A1 =

[-4477/10000,  2329486907257389/4503599627370496,     4782865323631331469319222111/3169126500570573503
[  4477/1000,                          -2069/400,
[          0, -2329486907257389/4503599627370496, -8089440504834163086702848799 51/3961408125713216879 67

lamba =

  -5.6313 + 0.0000i
  -0.0047 - 0.2491i
  -0.0047 + 0.2491i
```

modelParameters.m

```
delta = 0.4477;
N = 10;
c = 5.1725;
B = 0.35;
k = c/(N * 412)

% equilibrium point [1000, 0, 0]
tcellrhs = @(t,x)[k*x(2)*x(3) - delta*x(1); N*delta*x(1) - c*x(2); B*x(3) - k*x(2)*x(3)];
[t,x] = ode45(tcellrhs,[0,7],[1000,0,0]);

% equilibrium point [412, 120.2113, 138.89]
tcellrhs = @(t,x)[k*x(2)*x(3) - delta*x(1); N*delta*x(1) - c*x(2); B*x(3) - k*x(2)*x(3)];
[t,x] = ode45(tcellrhs,[0,7],[412,120.3112,138.89]);
```

modelParametersOutput

```
modelParameters

k =

    0.0013


t =

         0
    0.0000
    0.0000
    0.0000
    0.0000
    0.0000
    0.0000
    0.0000
    0.0000
    0.0000
    0.0000
    0.0000
    0.0000
    0.0000
    0.0000
    0.0000
    0.0000
    0.0000
    0.0000
    0.0000
    0.0000
    0.0000
    0.0001
    0.0001
    0.0001
    0.0002
    0.0004
    0.0005
    0.0007
    0.0009
    0.0018
    0.0026
    0.0035
    0.0044
    0.0088
    0.0131
```

0.0175
0.0219
0.0438
0.0657
0.0877
0.1096
0.1425
0.1754
0.2083
0.2412
0.2801
0.3190
0.3578
0.3967
0.4422
0.4876
0.5331
0.5786
0.6317
0.6848
0.7379
0.7909
0.8543
0.9177
0.9811
1.0444
1.1214
1.1984
1.2753
1.3523
1.4485
1.5447
1.6408
1.7370
1.8616
1.9862
2.1108
2.2355
2.4008
2.5661
2.7315
2.8968
3.0718
3.2468
3.4218
3.5968
3.7718
3.9468
4.1218
4.2968
4.4443
4.5918
4.7392
4.8867
5.0141
5.1415
5.2689
5.3963
5.5322
5.6682
5.8041

```
    5.9401
    6.1043
    6.2686
    6.4328
    6.5970
    6.6978
    6.7985
    6.8993
    7.0000


x =

   1.0e+03 *

    1.0000         0         0
    1.0000    0.0000         0
    1.0000    0.0000         0
    1.0000    0.0000         0
    1.0000    0.0000         0
    1.0000    0.0000         0
    1.0000    0.0000         0
    1.0000    0.0000         0
    1.0000    0.0000         0
    1.0000    0.0000         0
    1.0000    0.0000         0
    1.0000    0.0000         0
    1.0000    0.0000         0
    1.0000    0.0000         0
    1.0000    0.0000         0
    1.0000    0.0000         0
    1.0000    0.0000         0
    1.0000    0.0000         0
    1.0000    0.0001         0
    1.0000    0.0001         0
    1.0000    0.0001         0
    1.0000    0.0002         0
    1.0000    0.0003         0
    1.0000    0.0005         0
    0.9999    0.0006         0
    0.9999    0.0008         0
    0.9998    0.0016         0
    0.9998    0.0024         0
    0.9997    0.0031         0
    0.9996    0.0039         0
    0.9992    0.0078         0
    0.9988    0.0117         0
    0.9984    0.0155         0
    0.9980    0.0194         0
    0.9961    0.0383         0
    0.9941    0.0567         0
    0.9922    0.0747         0
    0.9902    0.0923         0
    0.9806    0.1738         0
    0.9710    0.2457         0
    0.9615    0.3090         0
    0.9521    0.3646         0
    0.9382    0.4356         0
    0.9245    0.4936         0
    0.9110    0.5406         0
    0.8976    0.5785         0
    0.8821    0.6134         0
```

```
0.8669    0.6395         0
0.8520    0.6584         0
0.8373    0.6716         0
0.8204    0.6812         0
0.8039    0.6857         0
0.7877    0.6862         0
0.7718    0.6837         0
0.7537    0.6780         0
0.7360    0.6700         0
0.7187    0.6601         0
0.7018    0.6491         0
0.6822    0.6350         0
0.6631    0.6201         0
0.6445    0.6048         0
0.6265    0.5893         0
0.6053    0.5707         0
0.5848    0.5522         0
0.5650    0.5340         0
0.5458    0.5163         0
0.5228    0.4949         0
0.5008    0.4742         0
0.4797    0.4543         0
0.4595    0.4352         0
0.4345    0.4117         0
0.4110    0.3894         0
0.3887    0.3683         0
0.3676    0.3483         0
0.3414    0.3235         0
0.3170    0.3004         0
0.2944    0.2789         0
0.2734    0.2590         0
0.2528    0.2396         0
0.2337    0.2216         0
0.2161    0.2048         0
0.1998    0.1893         0
0.1848    0.1752         0
0.1708    0.1620         0
0.1580    0.1496         0
0.1461    0.1382         0
0.1367    0.1296         0
0.1280    0.1214         0
0.1198    0.1135         0
0.1122    0.1062         0
0.1059    0.1004         0
0.1001    0.0948         0
0.0945    0.0896         0
0.0893    0.0846         0
0.0840    0.0796         0
0.0791    0.0749         0
0.0744    0.0705         0
0.0700    0.0663         0
0.0650    0.0616         0
0.0604    0.0573         0
0.0561    0.0532         0
0.0522    0.0494         0
0.0499    0.0472         0
0.0477    0.0452         0
0.0456    0.0432         0
0.0435    0.0413         0
```

```
t =

         0
    0.0049
    0.0099
    0.0148
    0.0198
    0.0445
    0.0692
    0.0940
    0.1187
    0.1563
    0.1938
    0.2314
    0.2689
    0.3118
    0.3546
    0.3974
    0.4402
    0.4907
    0.5413
    0.5918
    0.6423
    0.7023
    0.7623
    0.8223
    0.8823
    0.9556
    1.0289
    1.1022
    1.1755
    1.2674
    1.3593
    1.4512
    1.5431
    1.6629
    1.7827
    1.9025
    2.0223
    2.1847
    2.3472
    2.5096
    2.6720
    2.8470
    3.0220
    3.1970
    3.3720
    3.5470
    3.7220
    3.8970
    4.0720
    4.2470
    4.4220
    4.5970
    4.7720
    4.9121
    5.0522
    5.1922
    5.3323
    5.4481
    5.5638
```

```
    5.6795
    5.7953
    5.9194
    6.0435
    6.1677
    6.2918
    6.4395
    6.5872
    6.7349
    6.8826
    6.9120
    6.9413
    6.9707
    7.0000


x =

  412.0000  120.3112  138.8900
  411.1951  126.2699  139.0241
  410.3972  132.0605  139.1533
  409.6060  137.6876  139.2777
  408.8215  143.1553  139.3974
  404.9935  168.2509  139.9298
  401.3110  189.9419  140.3625
  397.7589  208.6462  140.7081
  394.3234  224.7421  140.9786
  389.2991  244.9539  141.2693
  384.4802  260.8363  141.4404
  379.8376  273.1509  141.5167
  375.3446  282.5904  141.5206
  370.3777  290.5717  141.4604
  365.5546  296.1187  141.3491
  360.8566  299.7075  141.2020
  356.2665  301.7821  141.0338
  350.9696  302.7693  140.8236
  345.7902  302.4803  140.6107
  340.7172  301.1987  140.4039
  335.7394  299.2079  140.2119
  329.9416  296.2275  140.0124
  324.2597  292.7119  139.8473
  318.6879  288.7998  139.7204
  313.2198  284.6531  139.6358
  306.6722  279.4447  139.5944
  300.2693  274.1082  139.6208
  294.0079  268.6951  139.7150
  287.8837  263.2954  139.8787
  280.3932  256.6412  140.1834
  273.1105  250.0854  140.5959
  266.0324  243.6344  141.1143
  259.1539  237.3371  141.7386
  250.4778  229.4090  142.7112
  242.1275  221.7444  143.8589
  234.0958  214.3297  145.1795
  226.3732  207.1934  146.6724
  216.3817  197.9870  148.9709
  206.9237  189.2468  151.5824
  197.9794  180.9454  154.5069
  189.5270  173.0971  157.7471
  180.9474  165.1416  161.5953
  172.8916  157.6574  165.8183
```

```
165.3356  150.6278  170.4233
158.2566  144.0323  175.4184
151.6342  137.8178  180.8119
145.4444  132.0224  186.6175
139.6651  126.6674  192.8494
134.2819  121.6515  199.5171
129.2857  116.8132  206.6281
124.6445  112.3964  214.2111
120.3356  108.5094  222.2897
116.3621  104.8523  230.8668
113.4261  101.9028  238.0938
110.6822   99.2208  245.6696
108.1197   96.8712  253.6089
105.7465   94.6607  261.9114
103.9319   92.8390  269.0479
102.2375   91.1586  276.4494
100.6606   89.6289  284.1232
 99.2025   88.2042  292.0731
 97.7720   86.7561  300.9112
 96.4742   85.4389  310.0836
 95.3071   84.2600  319.5995
 94.2722   83.1941  329.4652
 93.2158   82.0399  341.6698
 92.3451   81.0639  354.3968
 91.6596   80.2862  367.6613
 91.1661   79.6628  381.4724
 91.0918   79.5518  384.2816
 91.0252   79.4478  387.1130
 90.9662   79.3508  389.9667
 90.9151   79.2608  392.8428
```

stackedPlot.m

```
delta = 0.4477;
N = 10;
c = 5.1725;
B = 0.13;
K = c/(N*412)
% x -> T*, V, T
tcellrhs = @(t,x)[k*x(2)*x(3) - delta*x(1);
    N*delta*x(1) - c*x(2);
    B*(1000 - x(3)) - k*x(2)*x(3)];
[t,x] = ode45(tcellrhs,[0, 100],[0,0.001,1000])
subplot(2,1,1)
plot(t,x(:,3))
title('Uninfected T-cells','interpreter','latex','FontSize',16)
xlabel('Days','interpreter','latex','FontSize',14)
subplot(2,1,2)
plot(t,x(:,2))
title('Virions','interpreter','latex','FontSize',16)
xlabel('Days','interpreter','latex','FontSize',14)
```

stackedPlot13.m

```
k = c/(N * 412);
delta = 0.4477;
N = 10;
c = 5.1725;
B = 0.35;
tcellrhs = @(t,x)[B*(1000 - x(1)) - k*x(3)*x(1);
    k*x(3)*x(1) - delta*x(2);
    -c * x(3);
    N*delta*x(2) - c*x(4);
]
% T T_star V_I V_NI
[t,x] = ode45(tcellrhs,[0,30],[205,225,350,0])
subplot(2,1,1)
plot(t,x(:,1))
title('Uninfected T-cells','interpreter','latex','FontSize',16)
xlabel('Days','interpreter','latex','FontSize',14)
subplot(2,1,2)
plot(t,x(:,2))
title('Virions','interpreter','latex','FontSize',16)
xlabel('Days','interpreter','latex','FontSize',14)
```

References

[1] Alan S. Perelson, Avidan U. Neumann, Martin Markowitz, John M. Leonard, David D. Ho* (1996), *HIV-1 Dynamics in Vivo: Virion Clearance Rate, Infected Cell Life-Span, and Viral Generation Time*, American Association for the Advancement of Science.