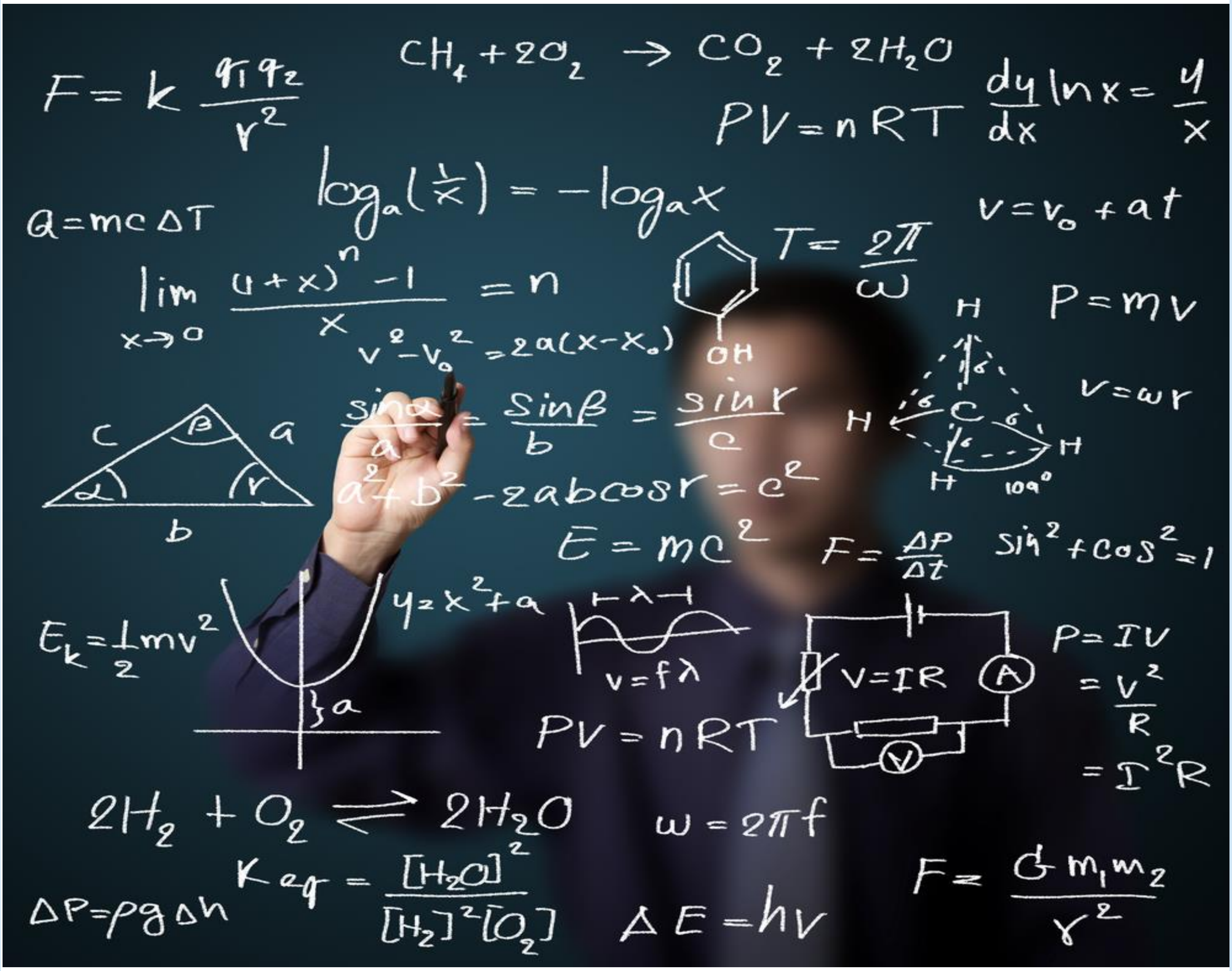


MT224 Semester Project

Syeda Ameera Haider (20I-0799), Naufil Moten (20I-0642), M. Ahmed Baig (20I-1884)
CS-B (Batch 2020)

Differential Equations

Table of Contents

Objectives and Introduction of the problem

Objective of our project is to create a MATLAB program for mathematical model of HIV dynamics using the equations and concepts of Differential Equations studied in our course.

We start our work by studying the Mathematical model of HIV dynamics given in the question, breaking down into simple processes, using the variables of population which is uninfected of CD4+ T cells, population which is infected of CD4+ T cells, and population at the time of virus. All of the variables are taken at time T.

Step-by-Step Analytical Solution

a) Let's assume treatment as given Set $B=0$

Differential Equations :

- $\frac{d}{dt}T(t) = \lambda - \delta T(t)$
- $\frac{d}{dt}I(t) = -\mu I(t)$
- $\frac{d}{dt}V(t) = N\mu I(t) - \gamma V(t)$

b) Reduced equations with initial conditions

$T(0) = t_0 \ , \ I(0) = I_0 , V(0) = V_0$

First solve for $I(t)$

$\frac{d}{dt}I(t) = -\mu I(t)$

$I(t) = ce^{-\mu t} \rightarrow$ (c is constant)

now use initial conditions

$c = I_0$

$I(t) = I_0 e^{-\mu t}$

Solve for $T(t)$

$\frac{d}{dt}T(t) = \lambda - \delta T(t)$

$\frac{1}{\lambda - \delta T(t)} dT = dt$

Integrate both sides

$\int \frac{1}{\lambda - \delta T(t)} dT = \int dt$

$-\frac{1}{\delta} \ln |\lambda - \delta T(t)| = t + c$

$$T(t) = \frac{\lambda}{\delta} - \frac{1}{\delta} e^{-(\delta t + \delta c)}$$

Initial conditions $T(0) = T_0$

$$T(0) = \frac{\lambda}{\delta} - \frac{1}{\delta} e^{-(\delta \times 0 + \delta c)}$$

$$T_0 - \frac{\lambda}{\delta} = -\frac{1}{\delta} e^{-\delta c}$$

So, $T(t) = \frac{\lambda}{\delta} + \left(T_0 - \frac{\lambda}{\delta}\right) e^{-\delta t}$

$$T(t) = (1 - e^{-\delta t}) \frac{\lambda}{\delta} + T_0 e^{-\delta t}$$

For $V(t)$,

$$\frac{d}{dt} V(t) = N \mu I(t) - \delta V(t)$$

Sub the value of $I(t)$

$$\frac{d}{dt} V(t) = N \mu I_0 e^{-\mu t} - \delta V(t)$$

$$\frac{d}{dt} V(t) + \delta V(t) = N \mu I_0 e^{-\mu t} \longrightarrow (1)$$

Integrating factor of equation

$$I = e^{\int \delta dt} = e^{\delta t}$$

Now multiply eq 1 by $e^{\delta t}$

$$e^{\delta t} \frac{d}{dt} V(t) + \delta e^{\delta t} V(t) = N \mu I_0 e^{-\mu t} \cdot e^{\delta t}$$

$$\frac{d}{dt}\left[V(t) \cdot e^{\delta t} \right] = N \mu I_0 e^{(\delta - \mu) t}$$

$$\frac{d}{dt}\left[e^{\delta t}V(t)\right] = N\mu I_o e^{(\delta-\mu)t}$$

Integrating both sides with t

$$e^{\delta t}V(t) = \frac{N\mu I_o}{\delta-\mu}e^{(\delta-\mu)t} + c$$

$$V(0) = V_0$$

$$c = V_0 - \frac{N\mu I_o}{\delta-\mu}$$

$$V(t) = \frac{N\mu I_o}{\delta-\mu}e^{-\mu t} + \left[V_0 - \frac{N\mu I_o}{\delta-\mu}\right]e^{-\delta t}$$

$$V(t) = \frac{N\mu I_o}{\delta-\mu}\left[\frac{1}{e^{\mu t}} - \frac{1}{e^{\delta t}}\right] + \frac{V_0}{\delta\mu e^{\delta t}}$$

c) Equations of $V(t)$

$$V(t) = \frac{N\mu I_o}{\delta-\mu}e^{-\mu t} + \left[V_0 - \frac{N\mu I_o}{\delta-\mu}\right]e^{-\delta t}$$

Let $A = \frac{N\mu I_o}{\delta-\mu}$ ~~A~~ and $B = V_0 - \frac{N\mu I_o}{\delta-\mu}$

Now

$$V(t) = Ae^{-\mu t} + Be^{-\delta t}$$

Take log

$$\ln[V(t)] = -\mu t - \delta t + \ln[B] + \ln[A]$$

$$\ln[V(t)] = t(\mu - \delta) + \ln[B] + \ln[A]$$

Now if we have a graph of $\ln V$ and 't' is a straight line with a slope $-\mu-\delta$

Graph of $\ln V$ is straight line with slope $-\delta$ or $-\mu$

e) During couple of years, mathematicians and researcher have paid a great deal of exertion to displaying and examining HIV elements inside the host. The CD4+ T cell is HIV's essential target. HIV is the infection that causes AIDs, a deadly disease. The cooperation between the host cells and HIV might be portrayed utilizing numerical models of HIV elements. Standard differential condition models both linear and nonlinear were the most widely recognized models introduced. The associations between weak cells, tainted cells, infections, and safe cells are the subject of these models. Basic HIV models have supported in the formation of a superior information on the sickness and the various pharmacological therapy alternatives accessible to battle it.

Step-by-Step and Line-by-Line MATLAB Solution

Instruction Manual

After starting MATLAB, we will open our file (i200799_i200642_i201884_MT224_Project_B.m), in the top right corner we will press run or alternatively we can also press F5 to run the program. The name of our course along with the name of the team members is displayed and then the three reduced equations for __ are displayed on output window, the graphs for the three equations are displayed on a different window and the program ends.

MATLAB Code Description

In our MATLAB code, **L** is $\boldsymbol{\lambda}$ which is a constant input source of uninfected cells per day (the human body produces these cells daily in the thymus), **A** is $\boldsymbol{\delta}$ which is normal loss rate constant of uninfected cells, $\boldsymbol{\mu}$ is **U** which is the loss rate constant of infected cells, **G** is $\boldsymbol{\gamma}$ which is loss rate constant of free virus and $\boldsymbol{N}$ is **N** which is number of virions produced per day per infected cell.

After this we derived the equations by studying the units of these systems. The equations are as follow:

$\boldsymbol{T}(\boldsymbol{t})$ which is the population of uninfected CD4+ T cells at time t

$\boldsymbol{I}(\boldsymbol{t})$ which is the population of infected CD4+ T cells at time t

$\boldsymbol{V}(\boldsymbol{t})$ which is the population of virus at time t

To get the reduced forms, we declared all the equations and their initial conditions and then solved them using the *dsolve* command. The outputs for all three equations were displayed. To plot these equations, we assumed the variables to be

λ = L = 5 cells.day^{-1}

δ = A = 5 day^{-1}

μ = U = 0.5 days^{-1}

N = N = 800 virions^{-1} days^{-1}

γ = G = 48 days^{-1}

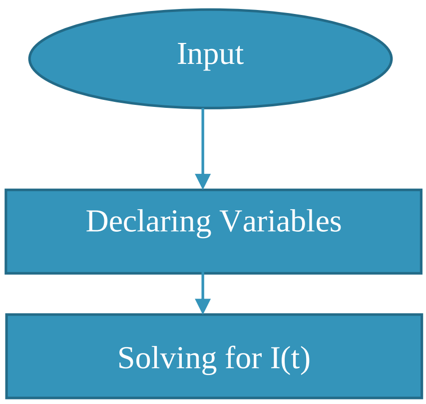
Io = 2

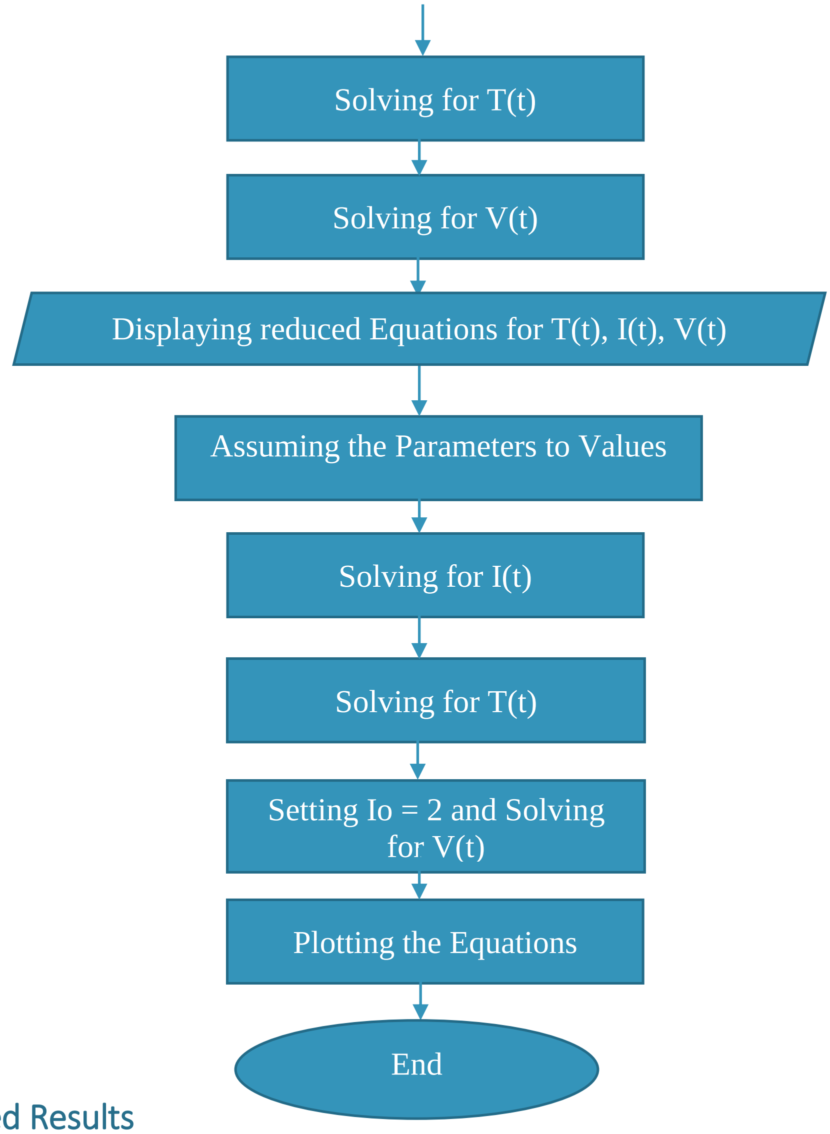
Then, we solved the equations again and plot the graphs using the *fimplicit* function of MATLAB.

Commands Used

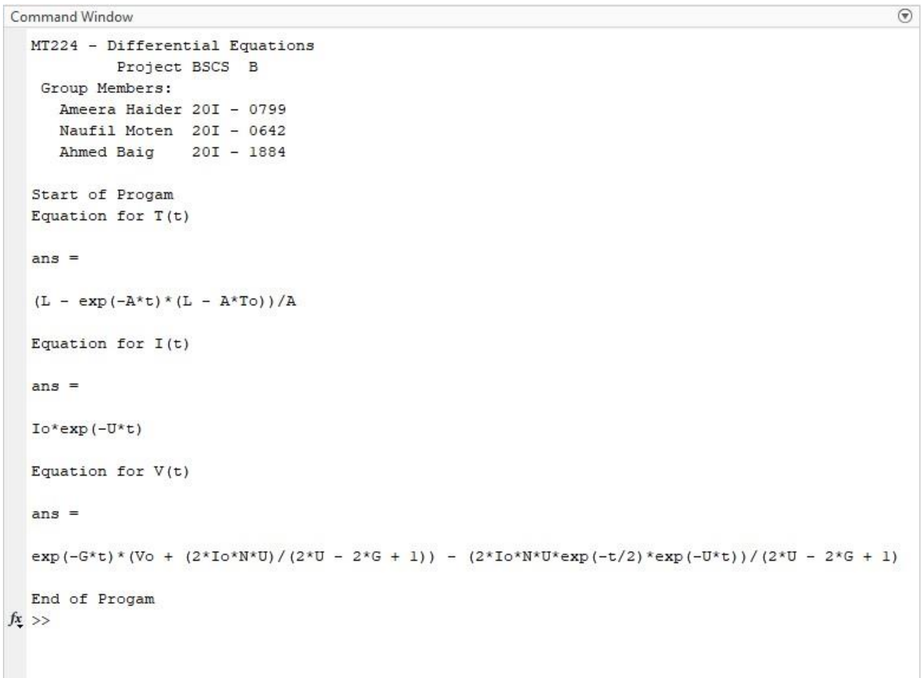
Command	Explanation
disp( ) / display( )	Output Command
syms variable;	Declaring the variables as Symbols
dsolve	computes symbolic solutions to ordinary differential equations
fimplicit	plots the implicit function defined by f(x,y) = 0 over the default interval [-5 5] for x and y

Flowchart of the solution

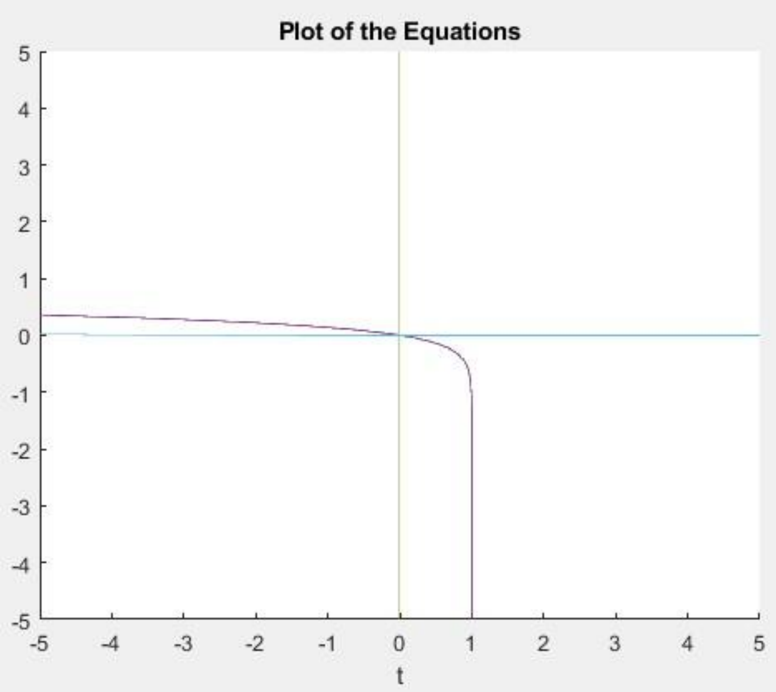




Detailed Results



```
Command Window
MT224 - Differential Equations
         Project BSCS  B
 Group Members:
   Ameera Haider 20I - 0799
   Naufil Moten  20I - 0642
   Ahmed Baig    20I - 1884

Start of Progam
Equation for T(t)

ans =

(L - exp(-A*t)*(L - A*To))/A

Equation for I(t)

ans =

Io*exp(-U*t)

Equation for V(t)

ans =

exp(-G*t)*(Vo + (2*Io*N*U)/(2*U - 2*G + 1)) - (2*Io*N*U*exp(-t/2)*exp(-U*t))/(2*U - 2*G + 1)

End of Progam
>>
```



Conclusion

Analysis: Using the concept of compartmental analysis we broke down the question into distinct stages and derived equations of each variable.

Methodology: We assumed the treatment as given set B = 0, and using the given initial conditions we took integral of the differential equations and solved them to get the reduced solution.

Comparison:

Hand written Solution	MATLAB Solution
$I(t) = I_0 e^{-\mu t}$	`Io*exp(-U*t)`
$T(t) = (1-e^{-\delta t})\frac{\lambda}{\delta} + T_0 e^{-\delta t}$	`(L - exp(-A*t)*(L - A*To))/A`
$V(t) = \frac{N \mu I_0}{\delta - \mu}\left[\frac{1}{e^{\mu t}} - \frac{1}{e^{\delta t}}\right] + \frac{V_0}{\delta \mu e^{\delta t}}$	`exp(-G*t)*(Vo + (2*Io*N*U)/(2*U - 2*G + 1))` `- (2*Io*N*U*exp(-t/2)*exp(-U*t))/(2*U - 2*G + 1)`

MATLAB Code of the Program

clear

```
clc

disp ("MT224 - Differential Equations");
disp ("        Project BSCS  B      ");
disp (" Group Members: ");
disp ("   Ameera Haider 20I - 0799");
disp ("   Naufil Moten  20I - 0642");
disp ("   Ahmed Baig    20I - 1884");
disp (" ");

%Start of the Solution
disp ("Start of Progam");

% T(t) is the population of uninfected CD4+ T cells at time ?
% I(t) is the population of infected CD4+ T cells at time ?
% V(t) is the population of virus at time t
% Inital Conditions
% T(0) = To
% I(0) = Io
% V(0) = Vo
% λ is L
% δ is A
% β is B
% μ is U
% γ is G
% N is N

syms T(t) TT(t) I(t) II(t) V(t) VV(t) Io IIo To TTo Vo VVo L A U N G

%Solving for I(t)
ode1 = (diff(I,t) == (-U.*I(t)));
cond1 = I(0) == Io;
I(t) = dsolve(ode1, cond1);
```

```
%Solving for T(t)
ode2 = (diff(T,t) == (L - A .* T(t)) );
cond2 = T(0) == To;
T(t) = dsolve(ode2, cond2);

%Solving for V(t)
ode3 = (diff(V,t) == (N.*U.*(I(t).*exp(-t/2)) - (G.*V(t)) ));
cond3 = V(0) == Vo;
V(t) = dsolve(ode3, cond3);

%Displaying General the Equation
disp ("");
disp ("Equation for T(t)");
display (T(t));
disp ("Equation for I(t)");
display (I(t));
disp ("Equation for V(t)");
display (V(t));

%Assuming the Parameters to be These Values
L = 5;
A = 5;
U = 0.5;
N = 800;
G = 48;

%Solving for I(t)
ode4 = (diff(II,t) == (-U.*II(t)));
cond4 = II(0) == IIo;
II(t) = dsolve(ode4, cond4);

%Solving for T(t)
ode5 = (diff(TT,t) == (L - A .* TT(t)) );
cond5 = TT(0) == TTo;
```

```
TT(t) = dsolve(ode5, cond5);

%Setting Io = 2 and Solving for V(t)
ode6 = (diff(VV,t) == (N.*U.*(2.*exp(-t/2)) - (G.*VV(t)) ));
cond6 = VV(0) == VVo;
VV(t) = dsolve(ode6, cond6);

%Plotting the Equations
title('Plot of the Equations');
xlabel('t');
hold on;
fimplicit(TT(t));
fimplicit(II(t));
fimplicit(VV(t));
hold off;

%End of the Solution
disp ("End of Progam");
```