ISC 4232 : Lab Assignment #3

Table of Contents

Lydia LaSeur October 7, 2014

Part 1: Modify Existing RK Library

Modify the previously completed explicit Runge Kutta library to handle a system of firt order IVPs. Run and test the convergence rates for stages 1 through 4 using the driver and the following system of three first order IVPs.

w1'(t) = 2*w2(t) - 4t

w2'(t) = -w1(t) + w3(t) - e^t +2

w3'(t) = w1(t) - 2*w2(t) + w3(t) + 4t

w1(0) = -1 ; w2(0) = 0 ; w3(0) = 2 ; 0 < t < 10

`RK_sysDriver`

```
1-STAGE RUNGE KUTTA METHOD FOR 3 EQN SYSTEM
Step Size    Abs Error at T      Convergence Rate
---------    --------------      ----------------
0.100000     0.2841184963409127   n/a
0.050000     0.1620903098136834  0.809694909866
0.025000     0.0873566863729663  0.891807807436
0.012500     0.0454653063712811  0.942152058187
0.006250     0.0231738148988156  0.972270512789
------------------END OF TABLE------------------


2-STAGE RUNGE KUTTA METHOD FOR 3 EQN SYSTEM
Step Size    Abs Error at T      Convergence Rate
---------    --------------      ----------------
0.100000     0.0115700211887305   n/a
0.050000     0.0030019646815499  1.946412597821
0.025000     0.0007673855934777  1.967883417735
0.012500     0.0001941677453643  1.982648116792
0.006250     0.0000487693707306  1.993256301384
------------------END OF TABLE------------------


3-STAGE RUNGE KUTTA METHOD FOR 3 EQN SYSTEM
Step Size    Abs Error at T      Convergence Rate
```

```
---------    --------------      ----------------
0.100000     0.0002790355842399   n/a
0.050000     0.0000361103203718  2.949965989559
0.025000     0.0000046163706598  2.967580248678
0.012500     0.0000005843165073  2.981937114198
0.006250     0.0000000734043713  2.992812161606
------------------END OF TABLE------------------


4-STAGE RUNGE KUTTA METHOD FOR 3 EQN SYSTEM
Step Size    Abs Error at T      Convergence Rate
---------    --------------      ----------------
0.100000     0.0000059906518559   n/a
0.050000     0.0000003899890277  3.941207554784
0.025000     0.0000000249883728  3.964104669610
0.012500     0.0000000015830650  3.980464571157
0.006250     0.0000000000992843  3.995011766087
------------------END OF TABLE------------------
```

Part 2: Model of Simple Viral Infection

(a). Model a simple viral infection using the following system and given parameters. Plot the density of the virus, v(t), and the density of antibodies, a(t) over the given domain, in time t.

a'(t) = k*v(t)

v'(t) = r*v(t) - p*v(t)a(t)

a(0) = 0 ; v(0) = .01 ; 0 < t < 96 (hours)

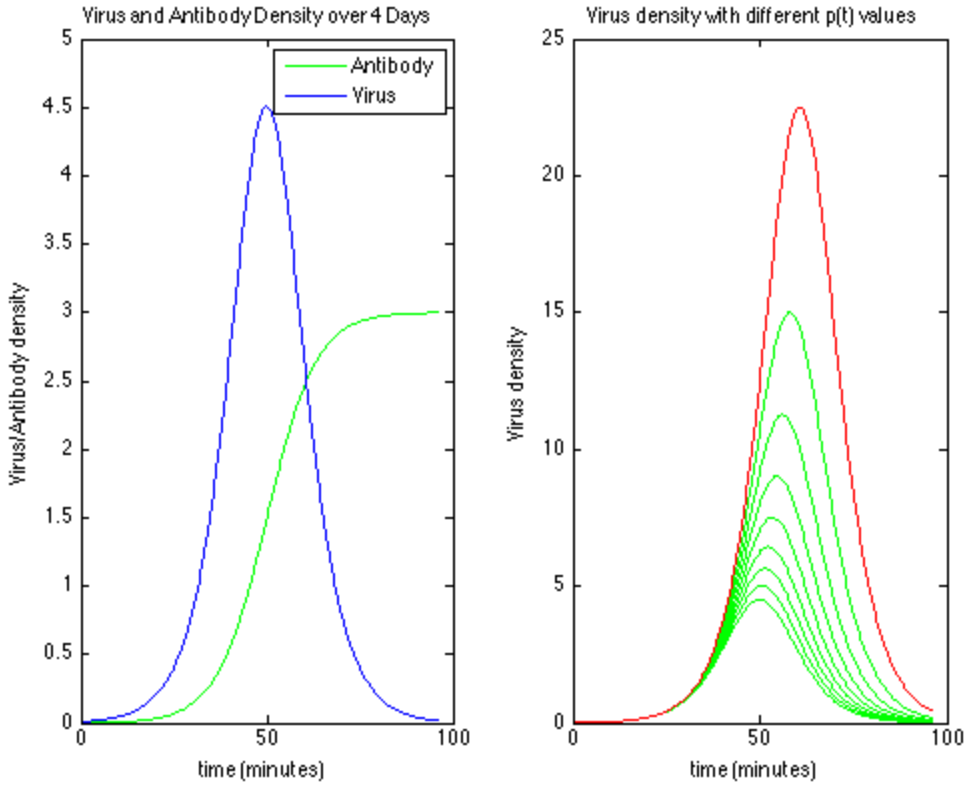
k = .025 ; r = .15 ; p = .1

Where k is the proportionality constant (number of antibodies produced is proportional to the number of viruses present). r is the rate at which new viruses occur. p is the probability that at any gven instance a particular antibody will encounted a particular virus and successfully destroy it. See both solutions on the first plot below.

(b). Decrease the value of p by .01 and run the model for each new value of p. PLot the solution of the viral denity for each run to determine when the patient will die, i.e. when the viral density exceeds 20. Keep all other parameters and initial conditions the same as before. See results in the second plot below.

`lab3_pt2`

```
Patient has died. Viral threshold of 20.000000 reached.
p(t) = 0.020000 at event of patient death
```



The color of the curve correponds to the status of the patient. Green means the patient survived and red means the patient died.

Part 3: Model Viral Infection with Single Mutation

Modify the previous model to account for mutation of the virus. To keep it simple, the virus will only mutate once. Since the virus will mutate different antibodies will be produced to fight the new mutated virus. This means two more DE mut be added to the system, one for the viral density of the mutated strain v2'(t), and one for the density of the antibodies that will fight the mutated virus a2'(t). Plus, a DE the describes the general immune response g'(t), i.e. production of antibodies to fight any virus, not a specific one. The 'strength' of the the general response is denoted by a constant, s. The virus is also able fight the antibodies. The strength of the virus is denoted by the constant u. The model is described by the five equation system below.

v1'(t) = v1(t) * (r - s*g(t) - p*a1(t)

a1'(t) = k*v1(t) - u*a1(t) * (v1(t) + v2(t))

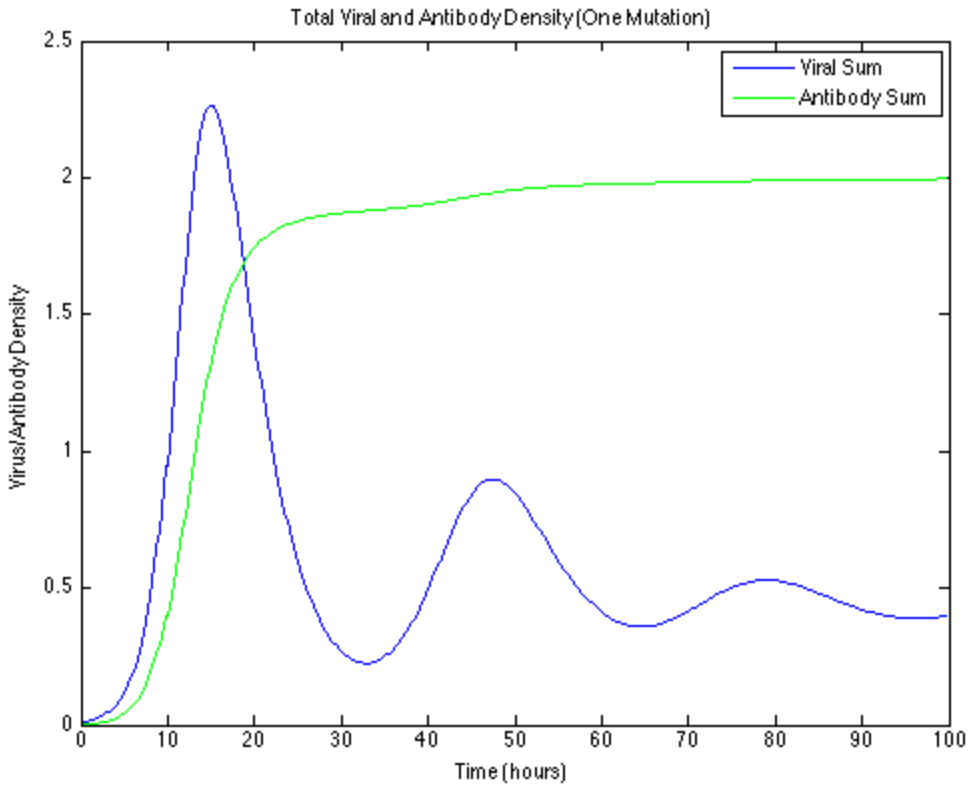
v2'(t) = v2(t) * (r - s*g(t) - p*a2(t)

a2'(t) = k*v2(t) - u*a2(t) * (v1(t) + v2(t))

g'(t) = (k - u*g(t)) * (v1(t) + v2(t))

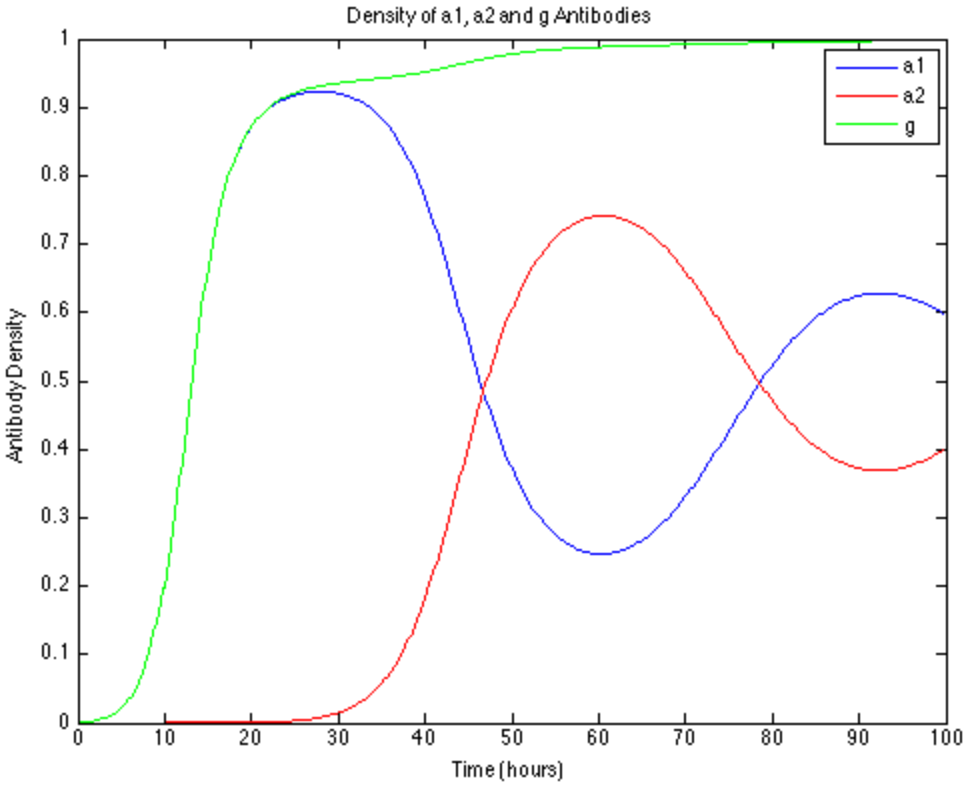
a1(0) = a2(0) = 0 ; v1(0) = .01 ; v2(0) = 0 ; g(0) = 0 ; 0 < t < 100

lab3_pt3



The plot above shows the total viral density and the total antibody density. The first big peak in the viral density is the mostly the original, non-mutated virus. At 10 hours, the virus mutates and second virus is introduced into the system. As the a1 antibodies fight the original virus the viral density begins the decrease but now the density of the mutated strain increases rapidly since the a2 antibodies haven't had time to replicate. This is the second peak in the viral density. The viral density will continue to oscillate in this fashion, while continuously decreasing. While the antibody sum never decreases the plot below shows that the antibody denisty for the specific anitboies, a1 and a1, do oscillate (due to the addition of the virus fighting back)but in a manner that they're behavoir mirrors each other, enuring that their sum doesn't decrese. In our model, the virus cannot fight the general antibodies, g, so that denisty never decreases. In this model using these parameters, the patient survives since the viral density does not exceed 20.

```
%plot the inidiviudal anutobdy densities
plot(soln(:,1),soln(:,3),'b') %a1(t) density
hold on
plot(soln(:,1),soln(:,5),'r') %a2(t) density
plot(soln(:,1),soln(:,6),'g') %g(t) desnity
xlabel('Time (hours)')
ylabel('Antibody Density')
title('Density of a1, a2 and g Antibodies')
legend('a1','a2','g')
```



Published with MATLAB® R2014a