Alex Klein
Gateway Computing
March 6, 2019
Project B
Tests Document

Testing 'ProjectB.m'

Test 1: Sub-Function TNext

- Inputs: (plugging arbitrary values) T = 30, V = 25
- Expected Results: TNext = 29.25
- Equation:

$$T(t+1) = T(t) - RI(t) = T(t) - 0.001\,T(t)V(t)$$

- Note: the sub function was put into another .m file so it could be called
- Results:

```
>> Tnext = NewT(30, 25)

Tnext =

   29.2500
```

- Outcome: the results are equal to the expected (previously calculated) results

Test 2: Sub-Function INext

- Inputs: (plugging arbitrary values) T = 30, I = 10, V = 25
- Expected Results: INext = 10.7
- Equation:

$$I(t+1) = I(t) + RI(t) - RD(t) = I(t) + 0.001\,T(t)V(t) - 0.005I(t)$$

- Note: the sub function was put into another .m file so it could be called
- Results:

```
>> Inext = NewI(30, 10, 25)

Inext =

   10.7000
```

- Outcome: the results are equal to the expected (previously calculated) results

Test 3: Sub-Function VNext

- Inputs: (plugging arbitrary values) T = 30, I = 10, V = 25, Q = 0.5
- Expected Results: VNext = 37.5
- Equation:

$$V(t+1) = V(t) + M(1-Q)\,RD(t) - RC(t) = V(t) + 10.0(1-Q)\,I(t) - 0.05V(t)T(t)$$

- Note: the sub function was put into another .m file so it could be called
- Results:

```
>> Vnext= NewV(30, 10, 25, 0.5)

Vnext =

   37.5000
```

- Outcome: the results are equal to the expected (previously calculated) results

Test 4: Sub-Function Increment

- Inputs: (plugging arbitrary values) T = 0.4, I = 3, V = 1.1, Q = 0.7
- Expected Results: TNext =0.3996, INext = 2.9854, VNext = 10.07
- Note: the sub function and three other sub functions TNext, INext, and VNext were put into another .m file so it could be called
- Results:

```
>> [Tnext, Inext, Vnext]= Increment(0.4, 3, 1.1, 0.7)

Tnext =

    0.3996


Inext =

    2.9854


Vnext =

   10.0780
```

- Outcome: the results are equal to the expected (previously calculated) results

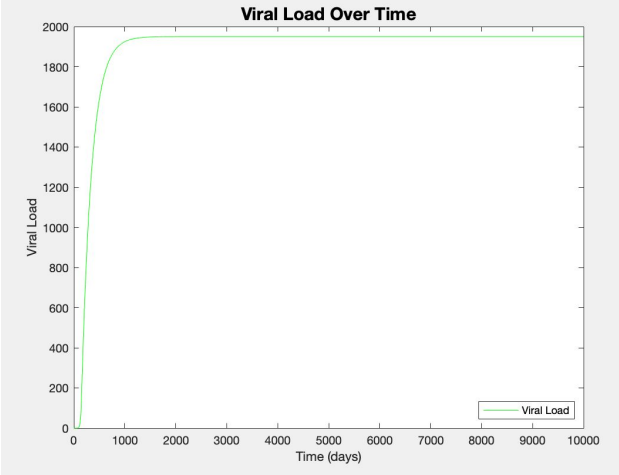
Test 5: No Drug Used

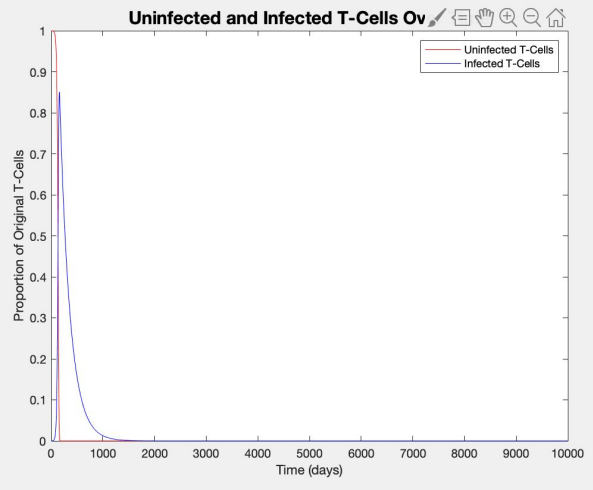
- Inputs: No drug used (Q = 0), normal viral load (V0 = 0.01), minT = 0 and maxtime = 10,000 so the model can be measured over a long period of time without interruption.
- Expected results: uninfected and infected T cell values (T and I) are close to 0, viral load (V) is high.
- Results:

```
>> ProjectB(0.01, 0, 10000, 0)

After 10000 days with a 0 percent effective drug:
The uninfected T-Cell count is 0.0000
The infected T-Cell count is 0.0000
The viral load is 1950.0100
```

- Outcome: the results are as we expected: T = 0, I = 0, V is a high value
- Graphical check: the graphs are expected:
    - Viral load starts at 0 and increases
    - T starts at 1 and decreases to 0
    - I starts at 0, increases to its max, then decreases to 0





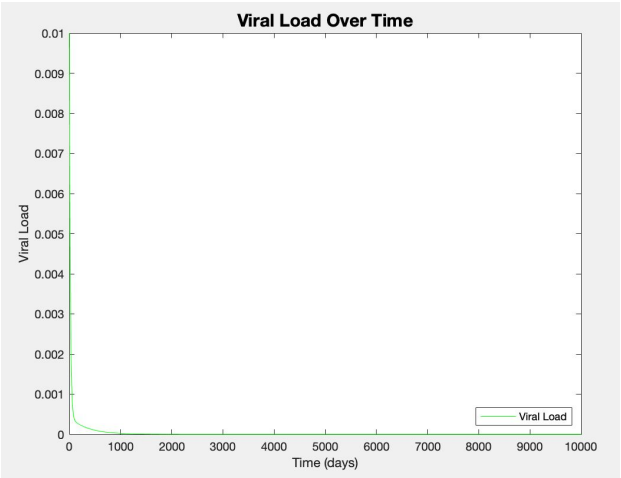
Test 6: Highly Effective Drug

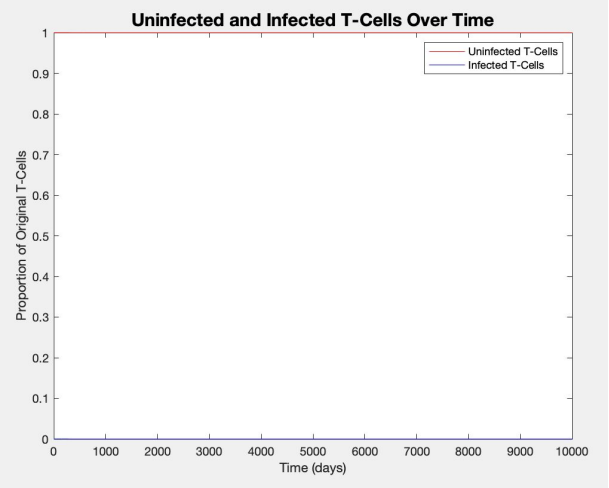
- Inputs: very highly effective drug used (Q = 0.99), normal viral load (V0 = 0.01), minT = 0 and maxtime = 10,000 so the model can be measured over a long period of time without interruption.
- Expected results: uninfected (T) cell value is close to 1, infected (I) cell value is close to 0, viral load (V) is close to 0.
- Results:

```
>> ProjectB(0.01, 0.99, 10000, 0)

After 10000 days with a 99 percent effective drug:
The uninfected T-Cell count is 0.9997
The infected T-Cell count is 0.0000
The viral load is 0.0000
```

- Outcome: the results are as we expected: T ~= 1, I = 0, V = 0
- Graphical check: the graphs are expected:
    - T remains at 1
    - I remains at 0
    - Viral load quickly decreases to 0





Test 7: Viral Load Comparison

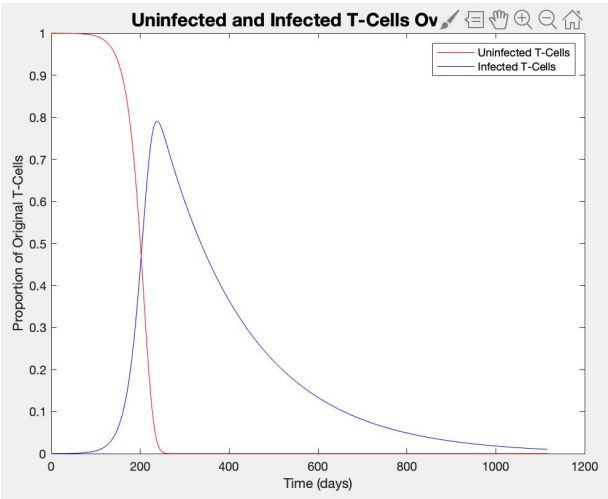
- Inputs:
    - (Test 7a): *Small* viral load (V0 = 0.01), average constant drug used (Q = 0.5) for both trials, minT = 0.01 for both trials, maxtime = 10,000 so the model can be measured over a long period of time without interruption.
    - (Test 7b): *Large* viral load (V0 = 1.0), average constant drug used (Q = 0.5) for both trials, minT = 0.01 for both trials, maxtime = 10,000 so the model can be measured over a long period of time without interruption.
- Expected results: the number of days before the simulation is stopped because T + I is less than minT should be less for Test 7b than Test 7a because the viral load is larger, and therefore HIV should spread faster.
- Results for Test 7a:

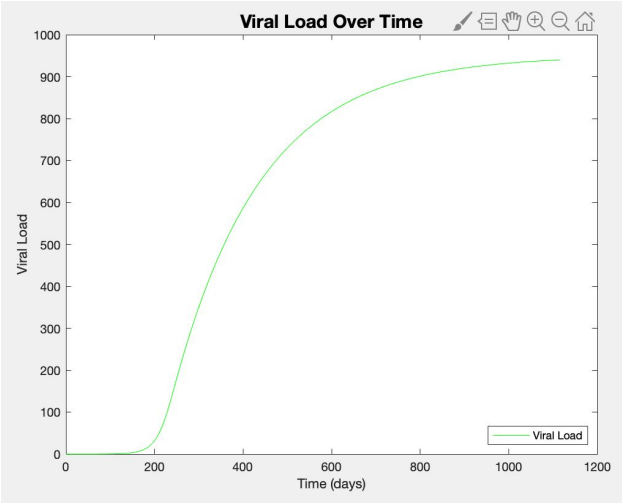
```
>> ProjectB(0.01, 0.5, 10000, 0.01)

The simulation was stopped after 1116 days because
the T cell count is less than the minT value of 0.0100

After 1116 days with a 50 percent effective drug:
The uninfected T-Cell count is -0.0000
The infected T-Cell count is 0.0100
The viral load is 940.0145
```

- Graphical check for Test 7a: the graphs are expected:
    - After 1116 days, T ~= 0, I ~= 0.01





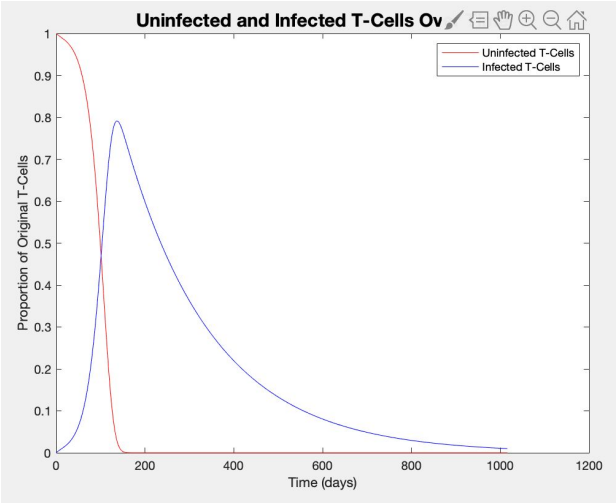
- Results Test 7b:

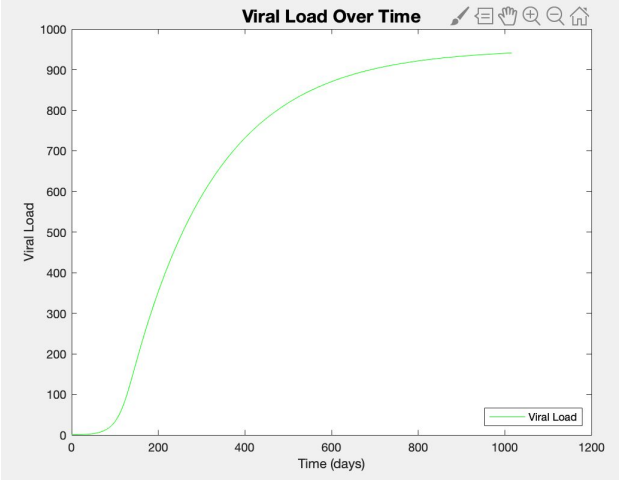
```
>> ProjectB(1, 0.5, 10000, 0.01)

The simulation was stopped after 1016 days because
the T cell count is less than the minT value of 0.0100

After 1016 days with a 50 percent effective drug:
The uninfected T-Cell count is 0.0000
The infected T-Cell count is 0.0100
The viral load is 941.0420
```

- Graphical check for Test 7b: the graphs are expected:
    - After 1016 days, T ~= 0, I ~= 0.01





- Outcome: the results are as we expected: Test 7b took fewer days (1016 days) than Test 7a (1116 days) to hit the minT exit command.

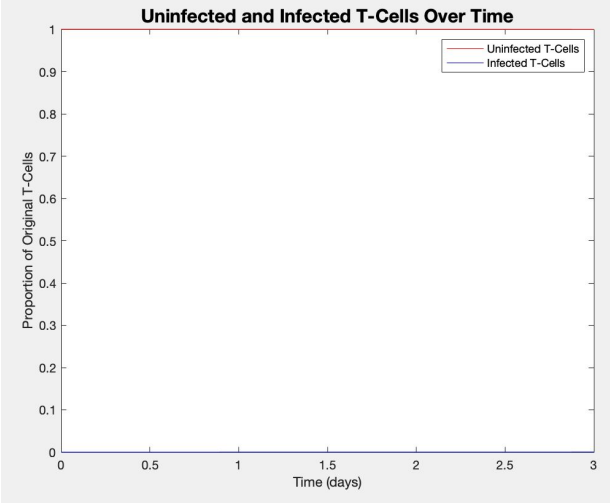
Test 8: Comparing with Computed Results

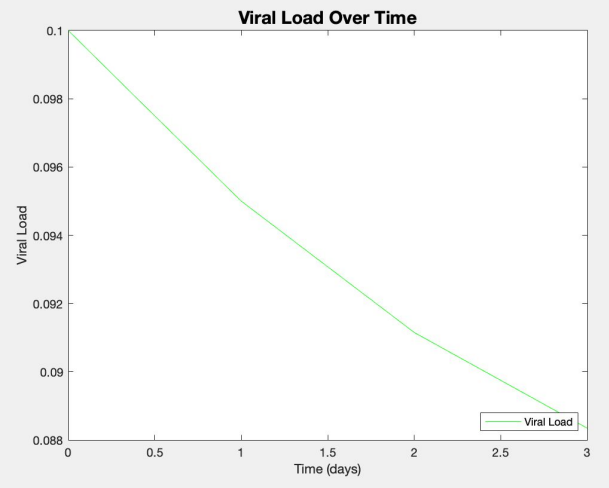
- Idea: will test the simulation for 3 days and compare the results of those that are produced by manually computing the answer.
- Inputs: maxTime = 3 in order to actually be able to manually calculate, Q = 0.1, Viral load = 0.1, minT = 0 (as this value will not be reached in any case).
- Results:

```
>> ProjectB(0.1, 0.1, 3, 0)

After 3 days with a 10 percent effective drug:
The uninfected T-Cell count is 0.9996
The infected T-Cell count is 0.0004
The viral load is 0.0865
```

- Graphs:





- Equations:

$$T(t+1) = T(t) - RI(t) = T(t) - 0.001\,T(t)V(t)$$

$$I(t+1) = I(t) + RI(t) - RD(t) = I(t) + 0.001\,T(t)V(t) - 0.005I(t)$$

$$V(t+1) = V(t) + M(1-Q)\,RD(t) - RC(t) = V(t) + 10.0(1-Q)\,I(t) - 0.05V(t)T(t)$$

- Results from manual computation:

Day	T	I	V
Start	1	0	0.1
0	0.9999	0.0001	0.095
1	0.9998	0.0002	0.0913
2	0.9997	0.0003	0.0886
3	**0.9996**	**0.0004**	**0.0865**

- Outcome: the results from Test 8 on the program match the manually computer results in the table.