Modeling Computer Virus Spread

Lucas Fraize, Sam Catalfo and Akhilesh Grandhi

November 2022

Introduction

The use of the internet has become an increasingly crucial part of our lives for over 30 years with the public release of the World Wide Web in 1991. It is primary form of communication and it is a requirement for the majority of population's livelihood in the U.S. at this point. So, an understanding of malicious activity on it is a necessity for us. In this paper, we explore the propagation of computer virus's across networks. We will use various methods and tools to create a model, analyze it, and apply it to real world examples.

Methods

The SI/SIR Model

The SIR model is a technique used frequently in epidemiology to analyze the spread of disease by compartmentalizing the population into three groups;

S: Susceptible to infection

I: Infected

R: Removed from susceptibility and infected

Traditionally, The SIR (Susceptible, Infected, Removed) model is described by the following system of ordinary differential equations (ODEs)

$$\frac{dS}{dt} = -\beta SI$$
$$\frac{dI}{dt} = \beta SI - \gamma I$$
$$\frac{dR}{dt} = \gamma I$$

where β represents the rate at which a susceptible members will become infected based upon contact with other infected members and γ represents the rate at which infected members will become removed and not susceptible again.

These same principles can be applied to modeling the spread of computer viruses within a given network. There is a key difference in our model compared to the traditional SIR model however. We assume there is a probability β that an infected system will infect a susceptible system with contact between the two and a probability γ that an infected computer will become susceptible again. In our case, the members return to being susceptible again once not infected anymore rather than removed. This is referred to as an SIS (Susceptible, Infected, Susceptible) model.

We also take other factors into consideration as well. As with most computer viruses, software updates remove computers from being susceptible to infection. We denote this probability in a given time step with ζ. We also take vitality dynamics into account as well with a probability of death μ of an infected system. Not that we are assuming that the given four probabilities do not change over time. We model this phenomena with the following system of ODEs

$$\begin{aligned}\frac{dS}{dt} &= -\beta SI + \gamma I - \zeta S\\ \frac{dI}{dt} &= \beta SI - \gamma I - \mu I\\ \frac{dR}{dt} &= \zeta S + \mu I\end{aligned}$$

Notice, even with taking vitality dynamics and removal within the susceptible population, that we can make this a formal SIS model as the recovered population R does not have any impact on the susceptible nor the infected populations. We can narrow the previous system down by removing R if it's results are not needed and we get

$$\begin{aligned}\frac{dS}{dt} &= -\beta SI + \gamma I - \zeta S\\ \frac{dI}{dt} &= \beta SI - \gamma I - \mu I\end{aligned}$$

Running our Models and Analysis

Phase Plane analysis

We only have one S-nullcline and two I-nullclines.

S-Nullcline	I Nullcline
$I = \frac{\zeta S}{\gamma - \beta S}$	$I = 0$
	$S = \frac{\gamma + \mu}{\beta}$

The S-nullcline is asymptotic with the S-asymptote at $S = \frac{\gamma}{\beta}$ and the I-asymptote at $I = 0$. So they only intersect in the scenario where there is a probability of death ($\mu > 0$). We will always have a steady state when both S and I are 0. If $\mu > 0$, then we will have a steady state only when $I < 0$.

Figure 1: In the sample image provided, we set $\beta = 0.1$, $\gamma = 0.8$, $\zeta = 0.2$, and $\mu = 0.2$.

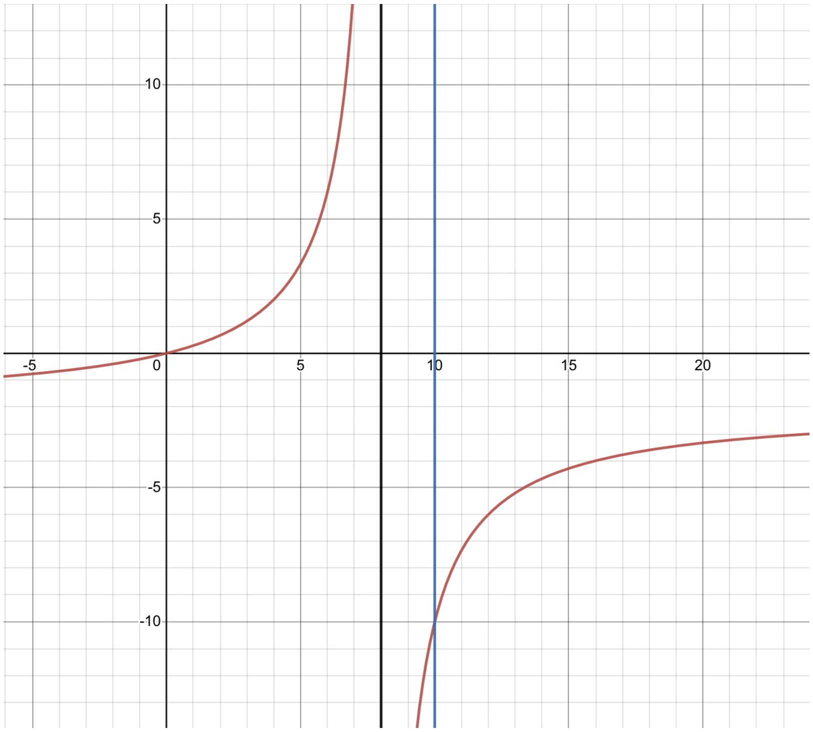
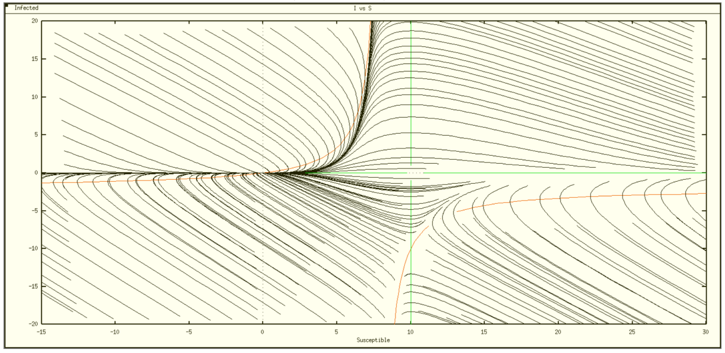


Figure 2: A sample phase plot done in XPP with the same parameters as before ($\beta = 0.1$, $\gamma = 0.8$, $\zeta = 0.2$, $\mu = 0.2$). We can see that the steady state at $(0, 0)$ is stable and the steady state at $(10, -10)$ is unstable

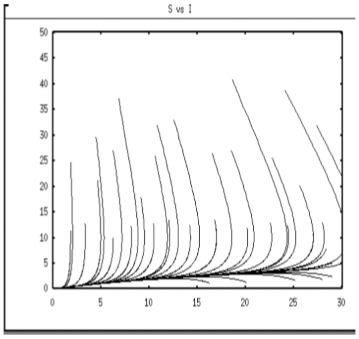


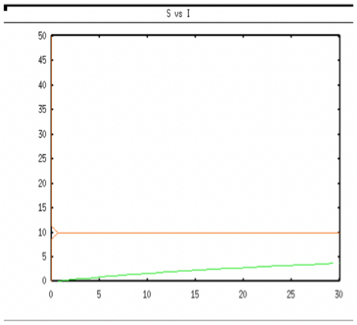
Graph Theory

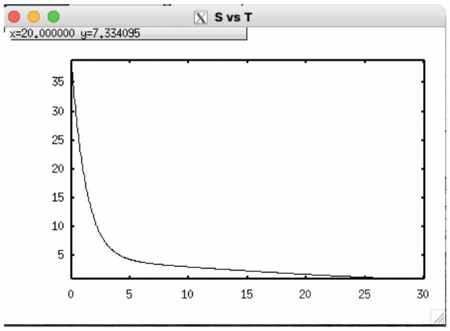
As computer viruses spread over networks, it is appropriate to add some graph theory to help understand it. We however will brush over this briefly. We ran multiple simulations in NetLogo to model the spread of viruses in K_{100} graphs (a network of 100 nodes such that they are all connected to one another) and the spread of viruses in non complete graphs.

Simulating real world examples

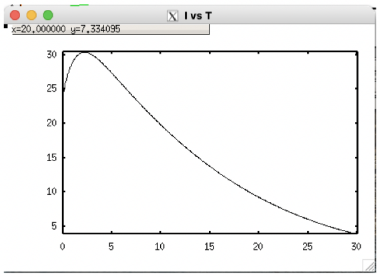
My Doom Virus







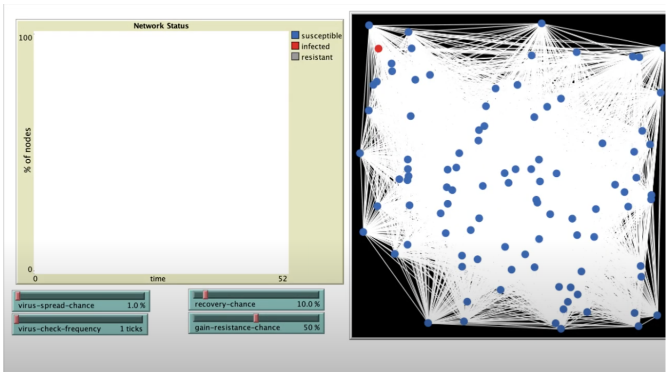
x=20.000000 y=7.334095



x=20.000000 y=7.334095

- These are the screenshots in XPP for the My doom virus.They are Phase Plane, Nullclines,the graph of S vs T and I vs T. Here we can see in the graph of S vs T, the graph shows that when the Time varies , the susceptible population decreases exponentially and is near to zero after certain amount time and if we check the graph of I vs T, the I increases at a point and when the time goes on it gradually decreases.The value of the parameters are $\beta = 0.01, \gamma = 0.1, \zeta = 0.5$ and $\mu = 0$.

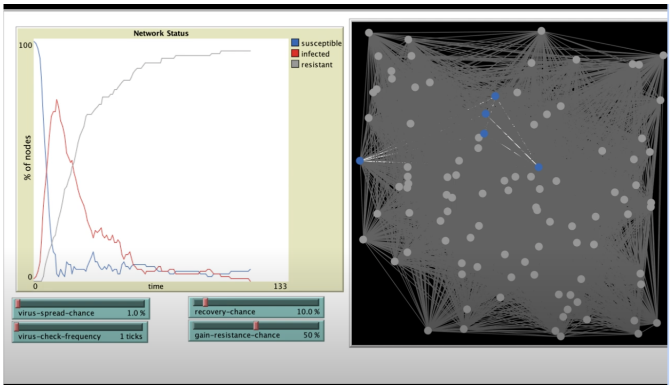
Modelling MyDoom in NetLogo



virus-spread-chance 1.0 %
virus-check-frequency 1 ticks
recovery-chance 10.0 %
gain-resistance-chance 50 %

- This is a screenshot of initial setup for one of the runs of the NetLogo simulation of our system, with K=100 initial nodes, and parameters matching those previously chosen for this model.

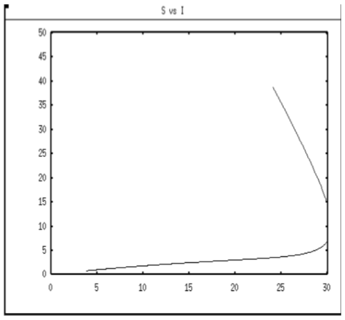
- We have chosen our parameters as these values to represent the system as accurately as possible. These values correspond to the parameters picked prior. Virus-Spread-Chance has been set to 1% because the way the virus spreads is if you click an executable from an unknown email, and that should be quite unlikely. We have then set the recovery-chance to 10% because this virus made it harder for an infected computer to patch itself by blocking access to Microsoft servers. We set the gain-resistance-chance to 50% because that is our estimated probability of a user installing the patch before becoming infected.

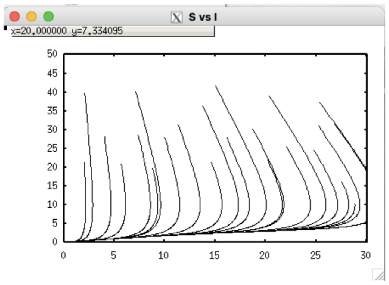


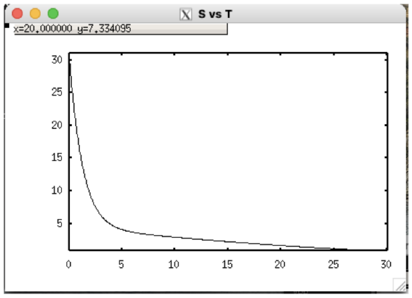
virus-spread-chance 1.0 %
virus-check-frequency 1 ticks
recovery-chance 10.0 %
gain-resistance-chance 50 %

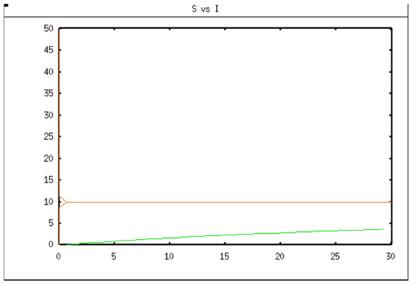
- This is one of the results we got after running this simulation. One can perceive that the results of the simulation line up with our prediction made in XPP. Specifically, the I vs T and S vs T graphs display similar behaviour in both the XPP and NetLogo simulations.

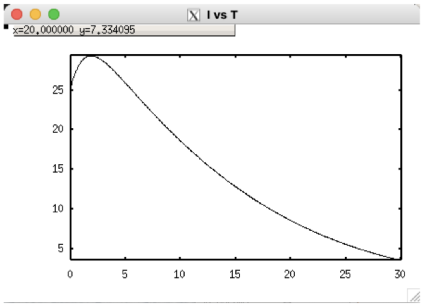
Samy Case





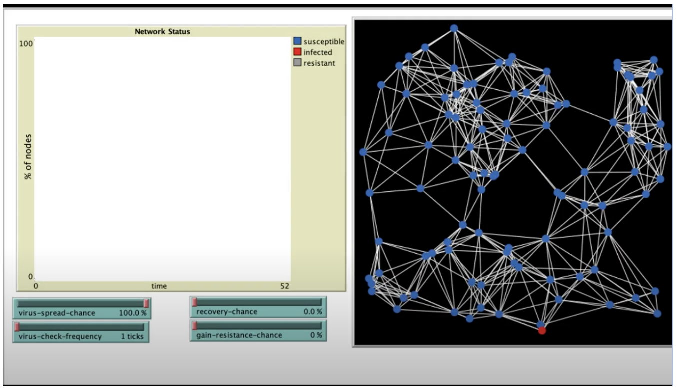






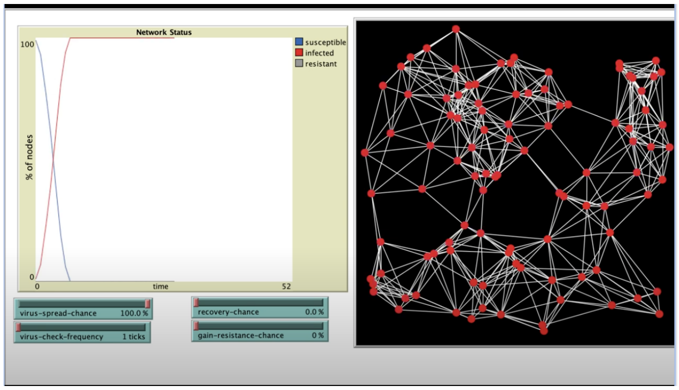
- These are the screenshots in XPP for the Samy Virus.They are Phase Plane, Nullclines,the graph of S vs T,I vs T and S vs I. Here we can see in the graph of S vs T, the graph shows that when the Time varies , the susceptible population decreases exponentially and is near to zero after certain amount time and if we check the graph of I vs T, the I increases at a point and when the time goes on it gradually decreases.The value of the parameters are $\beta = 1, \gamma = 0, \zeta = 0$ and $\mu = 0$.

Modelling Samy in NetLogo



virus-spread-chance 100.0 %
virus-check-frequency 1 ticks
recovery-chance 0.0 %
gain-resistance-chance 0 %

- This is a screenshot of initial setup for one of the runs of the NetLogo simulation of our system, with K=100 initial nodes, and parameters matching those previously chosen for this model. What you will notice is different is that there are only six connections between each node as opposed to every node being interconnected. This was done in order to mimic the social network MySpace where this virus was introduced to the world.



virus-spread-chance 100.0 %

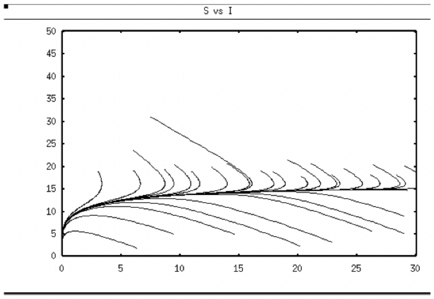
virus-check-frequency 1 ticks

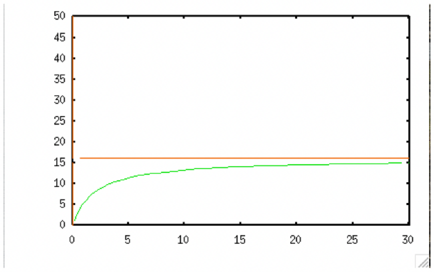
recovery-chance 0.0 %

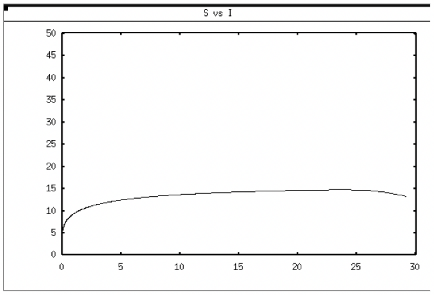
gain-resistance-chance 0 %

- Given the parameters we have chosen for this simulation, it should come as no surprise that we see exponential growth of the infected population. This was also reflected by the actual virus when it infected MySpace in 2004. The virus was the fastest spreading computer virus of all time, and the site shut down in order to patch it after it infected 1,000,000 people within just 24 hours.

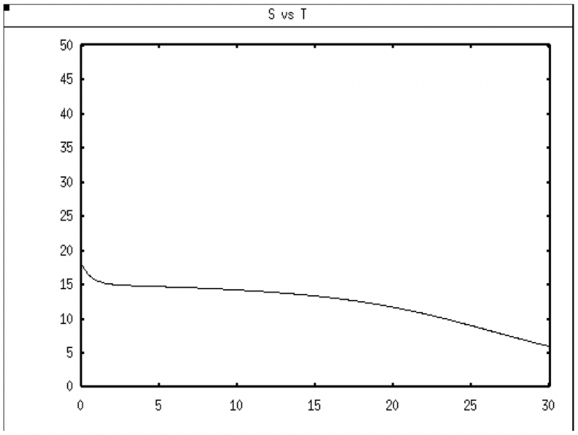
Average Computer Virus (1st part)

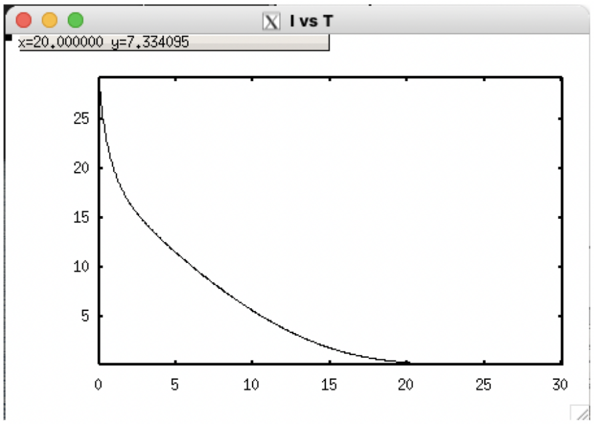






Average Computer Virus (2nd Part)

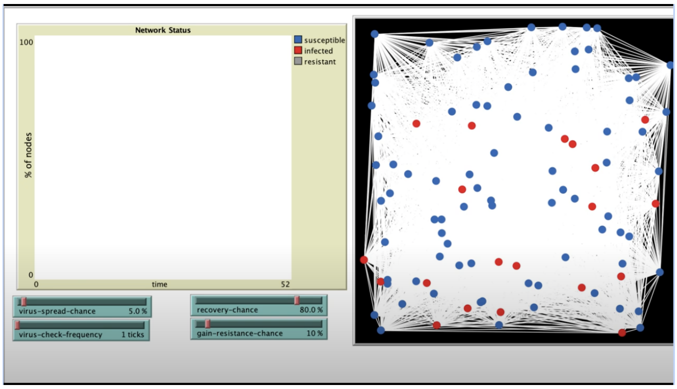




x=20.000000 y=7.334095

- These are the screenshots in XPP for the Average Computer Virus.They are Phase Plane, Nullclines,the graph of S vs T,I vs T and S vs I. Here we can see in the graph of S vs T, the graph shows that when the time varies , the infected population decreases exponentially and is near to zero after certain amount time and if we check the graph of S vs T, the S starts constantly and then it gradually decreases.The value of the parameters are $\beta = 0.05, \gamma = 0.8, \zeta = 0.10$ and $\mu = 0.01$.

Modelling Average Computer Virus in NetLogo



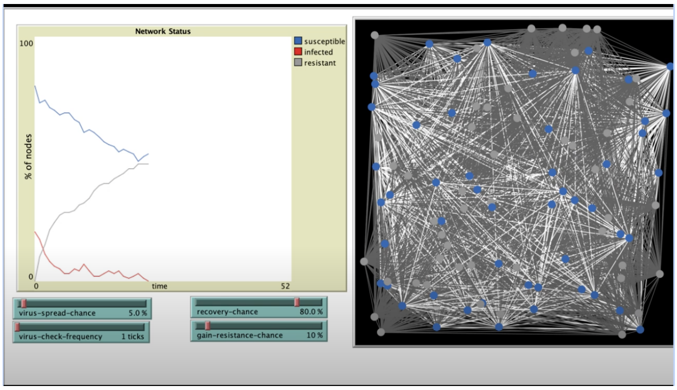
virus-spread-chance 5.0 %

virus-check-frequency 1 ticks

recovery-chance 80.0 %

gain-resistance-chance 10 %

- This is a screenshot of initial setup for one of the runs of the NetLogo simulation of our system, with K=100 initial nodes, and parameters matching those previously chosen for this model.



virus-spread-chance 5.0 %

virus-check-frequency 1 ticks

recovery-chance 80.0 %

gain-resistance-chance 10 %

- As we can see by the results of our simulation, modern day computer viruses tend to not last as long, or cause as much harm due to the evolution of technology throughout recent years heavily modifying the parameters of the system.
- These results also show that our XPP system was also accurate in modeling these parameters.

Conclusions and Further Discussion

Over the years, the behavior of computer viruses has changed as a result of more frequent updates and better security throughout all computer systems and networks. There are a multitude of ways we could have taken our models further. We could have expanded our systems to take into account the fact that our parameters such as the probability of infection and probability of recovery change over time. We could have also added a step function to help model the behavior of a virus before and after software updates that would remove computers from being susceptible. We also could have taken further steps with modeling the networks they propagate through as most computer networks are not complete graphs and have other key components to their structure we did not take into account.