An Agent-Based Approach to Modeling Ebola Outbreak

Introduction

The 2014-2016 Western Africa Ebola epidemic marked the largest outbreak of the disease ever recorded. During this time, some 28,616 confirmed or suspected cases were reported across Sierra Leone, Guinea, and Liberia. Even with the intervention of concerted humanitarian aid and directed relief efforts, 11,310 people in total died as a result.[1] As a result the World Health Organization (WHO) named Ebola an emerging disease likely to induce a major epidemic in December 2015.[2]

In order to learn what fundamental attributes drive the spread of disease, it is beneficial to create models that simulate real-life events. A common model of disease spread is the SEIR model. SEIR models work by categorizing members of some population as either *S*usceptible, *E*xposed, *I*nfected, or *R*emoved. By defining rates by which members transition between each of these categories and altering the parameters that affect these rates, we can observe how a population exposed to disease changes over time. One such parameter is the reproduction number, R_0, the average number of secondary cases spawned by a primary case. In Ebola, this measure has been estimated to be two, meaning an infected individual will pass the disease to two others.[3] SEIR models are often described by ordinary differential equations (ODEs), and related work of this nature is discussed further below.

One drawback of standard ODE-based SEIR models is that they do not directly account for the stochastic nature by which infected members of a population can interact with each other. As infected individuals travel, they risk spreading disease among the rest of the population. While the disease progression for Ebola is relatively quick and dramatic, the initial symptoms are fairly benign, such as fever and fatigue. Thus it is conceivable that there exists a window whereby symptomatic individuals could still interact with others before becoming immobile. Another drawback with ODE-based SEIR models is that they do not account for spatial constraints that could impact disease spread. Chowell et. al. (2015) performed statistical analyses on data generated during the West African Ebola outbreak and found that disease spreads at a polynomial rate on a local level, but exponentially at a broader global scale.[4] This suggests that there could be some underlying phenomena related to the way populations are distributed in space that could affect disease transmission, and merits further investigation.

This work presents the design and implementation of a novel agent-based model of Ebola disease spread that aims to improve upon traditional SEIR models by incorporating stochasticity and spatial constraints that are motivated by real-life conditions. In our model, virtual Agents inhabit nodes of a connected grid and may move randomly to a neighboring node at each time step. By ascribing each Agent one of the SEIR conditions and allowing them to change state based on their interactions with other agents

at a common node, we can connect emergent properties that arise in our model to real-world interactions. On the whole, we would like to determine how effectively our model can emulate the spread of Ebola.

In constructing a new model, there are a number of considerations we must make to evaluate its validity. First, we want to make sure that it correctly captures known behaviors. As an example, if we increase the disease model's reproduction number, R_0, the model should predict that people not only become infected at a quicker rate but are also removed much faster. Furthermore, we want to test the model's robustness by observing its behavior as we vary input parameters, such as the probabilities that Agents move away from a node, or the times for which an Agent inhabits a given SEIR condition. This is important since a model that is too sensitive may have too small a parameter search space to yield biologically significant results and, conversely, a model that lacks sensitivity may never demonstrate meaningful behaviors.

Finally, we want to utilize our choice of model parameters to elucidate behaviors associated with constraining the movement of Agents as well as constraining the grid's structure. These two constraints are the best means by which we attempt to transform our abstract model into a clearer picture of reality. Constraining infected Agent movement is analogous to the relative immobility of sick individuals, and constraints on the graph's connectivity impose barriers to movement and add a layer of nonuniformity more representative of the real world. By comparing our model with and without these constraints, we can determine whether or not Agent-based models can uncover large-scale behavior of disease spread.

Related Work

Current publications[5,6,7,8,9] studying the Ebola outbreak in West Africa during 2014-2015 have focused on using ODEs to describe SEIR models to examine the spread of the disease. Morrow applied the SEIR model, and then introduced criteria to model the effects of quarantine and vaccination, by the addition of graphical constraints by connecting nodes with the removal of nodes to represent quarantined zones, and vaccination rates modifying the ODEs for the simulation.[5] Ngwa and Teboh-Ewungken used a much more detailed ODE model that, among other things, accounted for uncertain cases, differential treatment of infected individuals, and allowing for deceased individuals to continue infecting new people under certain conditions.[6] Their analysis found a critical threshold parameter, R_0, which determined if the spread of Ebola reached a steady state. Notably, R_0 can be influenced by various interventions, including treatment and quarantine. If these interventions were enacted in a prompt fashion, they are sufficient to stop the spread of the disease.[6] Kiskowski adapted the SEIR model to account for different rates of disease spreading due to differences present at individual, household, and community scales.[7] Kiskowski's model showed that the disease could be characterized by a short exponential growth period, followed by a linear

acquisition of new cases, with overall exponential growth occurring due to the seeding of a new outbreak in a different community.[7]

Interestingly, Bartlett et al. used a different approach to study the outbreak. Instead of using ODEs to determine the SEIR model, they used a discrete age-structured model that probabilistically moved individuals into different groups depending on the total time that they had been infected.[8] Lastly, del Rey and Sanchez used a two-dimensional cellular automata model to model the spread of a virus through a population of cell phones via Bluetooth.[9] This approach uses agents to represent an individual cell phone, and then transitions each agent between susceptible, carrier, exposed, or infectious based on their proximity with other agents in a connected grid of nodes.[9] Note that each node can hold multiple agents at any given time step and that this type of simulation allows for the tracking of "real-time" information about the system as a whole, and on each individual involved in it. These studies inspired our own adaption of a cellular automata model to model the 2014 Ebola outbreak, as well as the ideas for enacting a more complex modeling of the behavior of agents through constraining the network, and quarantine of infectious individuals through the restriction of their movements.

Methods

Agents can have any one of four states at a given time step:

- Susceptible (*S)*: has not come into contact with an Infected Agent,
- Exposed (*E*): has come into contact with an Infected Agent but is not yet infectious,
- Infectious (*I*): able to traverse the graph and infect others, or
- Removed (*R*): can no longer infect others or become Infected; representative of persons who have either died or no longer be infected by with other Agents in the model.

Graph Construction

Our graphical model mimics the geographic size and population density of Conakry, Guinea, a West African city strongly impacted by the 2014 Ebola outbreak and for which the WHO has weekly reports of outbreak data.[10] We accomplished this by initializing 1.6 million Agents evenly across 196 nodes arranged in a square 14x14 grid (**Figure 2A**), as Conakry is home to about 1.6 million people with a land area of about 170 square miles. Thus, a single node conveniently represents about 1 square mile.

Algorithm

The algorithm initializes all Agents in state S except for a single Infectious Agent at node `(0,0)`. Next, the program alternates between updating each Agent's state and position for 3000 time steps (**Figure 1A, B**). At each time step, we also record the number and location of Agents in each state for later analysis.

An Agent's state is updated or held the same according to the rules defined in **Table 1**. The $S \rightarrow E$ transition probability for an Agent is described in two terms. The first term represents the fraction of

Agents at that node who are Infected, while the second term represents the probability of an Infectious Agent causing someone to become infected on a given time step. $|X|$ is the number of Agents with state X at a given node. R_0, the reproduction number, is the number of Agents that get infected by a single Infectious Agent before that agent becomes Removed. $T_{I\to R}$ is the time it takes for an Infectious Agent to become Removed.

Agents become Infectious if the time that they have been Exposed, t_E, exceeds the $E \to I$ time limit parameter, $T_{E\to I}$. Similarly, Agents become Removed if the time they have been Infected, t_I, exceeds the $I \to R$ time limit parameter, $T_{I\to R}$.

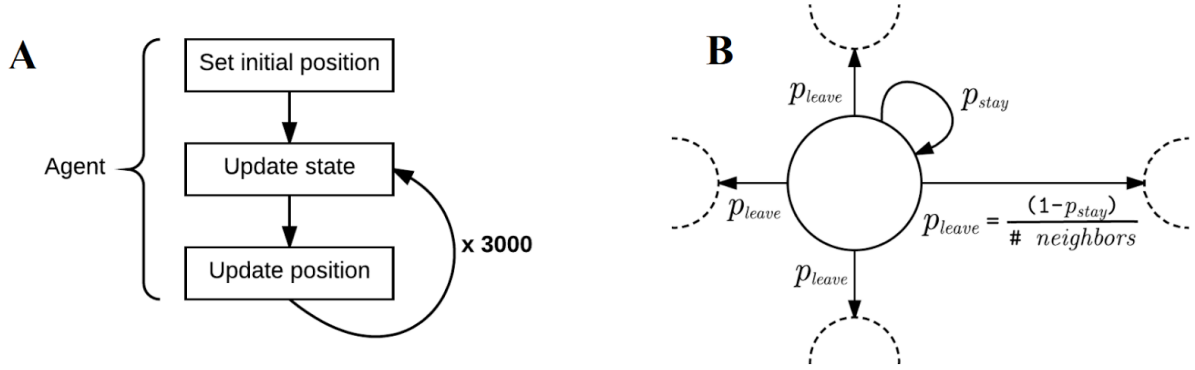

Figure 1. A) Simulation algorithm for a single Agent. Agents are initialized uniformly, then randomly move and interact with other Agents throughout the grid. **B)** An Agent will randomly decide to stay at its current position with probability p_{stay} and may move to a neighboring position with probability p_{leave}.

State Transition	Transition Probability
$S \to E$	$\frac{\lvert I\rvert}{\lvert I\rvert+\lvert E\rvert+\lvert S\rvert} \times \frac{R_0}{T_{I\to R}}$
$E \to I$	$\begin{cases} 1 & \text{if } t_E > T_{E\to I} \\ 0 & \text{otherwise} \end{cases}$
$I \to R$	$\begin{cases} 1 & \text{if } t_I > T_{I\to R} \\ 0 & \text{otherwise} \end{cases}$

Table 1. At every time step, each Agent updates its state according to the above state transitions and probabilities.

Experimental Conditions

Because we want to model the spread of Ebola in conditions that may more accurately account for its slow growth in local geographical settings, we chose to test the effects of two significant constraints on our simulation. The first, *Constrained Movement* (*CM*), ensures that Infected Agents are less likely to move as a result of their physiological symptoms. This is implemented by manipulating the parameter p_{stay} for Infectious Agents (**Figure 1B**). The second constraint, *Constrained Network* (*CN*), imposes restrictions on

the graph itself by eliminating predefined edges. This separates the graph into four distinct quadrants representing different localities or neighborhoods within a city (**Figure 2B**). This paper refers to the activation or inactivation of the constraints by giving them the values of True or False respectively.

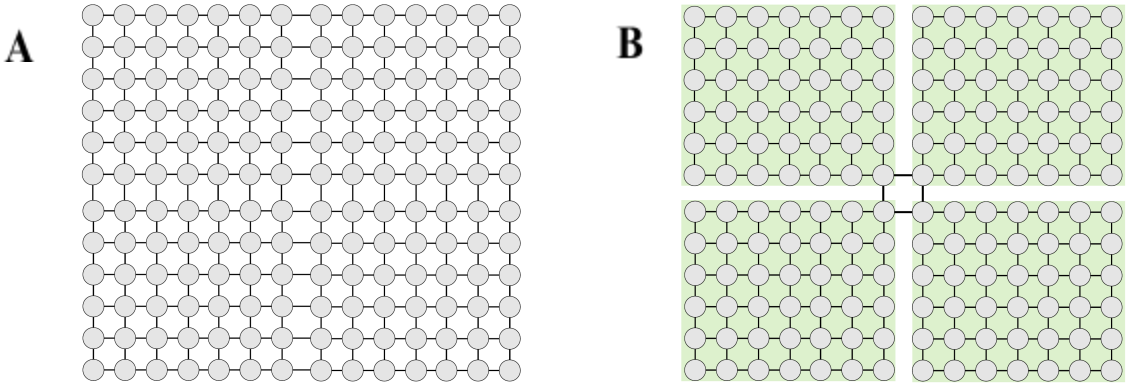

Figure 2. A) The unconstrained network (CN = False). **B)** The constrained network separated into four quadrants, with central edges connecting adjacent quadrants (CN = True).

In addition to these fundamental constraints, we also investigated our model's behavior while varying several other potentially important parameters. **Table 2** lists each parameter tested, the values used, and a short description of their role in the simulation. Default values for the first three parameters listed were all obtained directly from WHO reports. All other values tested for a given parameter were chosen arbitrarily to be much larger or smaller than the default to elicit changes in the simulation's behavior.

Parameter	Default Value	Other Values Tested	Description
$T_{E\to I}$	`randint(1,22)`	`randint(1,6)` `randint(1,51)`	Time before an Agent transitions $E \to I$.[11]
$T_{I\to R}$	8	2 20	Time before an Agent transitions $I \to R$.[3]
R_0	2	1 10	The expected number of secondary cases spawned by an Infectious Agent.[3]
p_{stay} ; CM = False	$\frac{1}{\#\ neighbors + 1}$	None	When movement is unconstrained, an Agent decides to move to its current position or any of the neighboring positions with equal probability.
p_{stay} ; CM = True	90%	50% 99.9%	When movement is constrained, an Agent decides to stay at its current position with probability p_{stay} but may choose to move to a neighboring position with $p_{leave} < p_{stay}$. p_{leave} is defined in **Fig1B.**
CN	False (**Fig2A**)	True (**Fig2B**)	Alters the graph's connectivity.

Table 2. Summary of parameters. `randint(x,y)` provides a random integer in the range `[x, y)`.

Model Assumptions

The model implementation described above makes the following assumptions:

1. At the start of the simulation, Agents are uniformly distributed across all nodes. A single Infectious Agent is placed at the bottom-left node.
2. During each step of the simulation, Agents perform a random walk to one of four neighbors in the cardinal directions with the ability to stay in place. If an Agent has less than 4 neighbors, then the probability of moving to those neighbors increases accordingly.
3. Because the parameters $T_{E\to I}$, $T_{I\to R}$, and R_0 are reported by WHO in units of days, we assume that time steps are equivalent to days.

Software Engineering

In order to precisely control all the parameters in the experiment and maximize our model's flexibility, we decided to build a software modeling suite from the ground up rather than use existing tools. Both the simulation and its analysis were completed in Python with aid of the Numpy and Matplotlib libraries (see Code Supplement). In addition to controlling the events in the simulation, the code in `Agent.py` contains the definition of an Agent as well as the methods for that Agent to move, change state, and interact with other Agents. The script takes the following as its arguments: [`int` # of time steps, `float` p_{stay}, `int` $T_{I\to R}$, `int` R_0, `int` $T_{E\to I}$, `boolean` *CM*, `boolean` *CN*, `string` output directory]. To maintain reproducibility across simulations, we used the number 7 to seed Numpy's random number generator.

Results

Figures 3 and **4** (attached at the end of this document) summarize much of our findings. **Figure 3** can be interpreted as a plot of the rate of infection over time. In contrast, because **Figure 4** plots the sum of Infected and Removed Agents per time step, it can be interpreted a plot of the cumulative amount of Agents affected by Ebola over time. Each figure contains plots of the nine different parameters settings we tried: the default parameters (top-left plots of **Figures 3** and **4**), and two parameter values for each of $T_{E\to I}$, $T_{I\to R}$, R_0, p_{stay} (all eight other plots). Each plot represents one choice of parameter settings and contains within it four trajectories: one for each possible *CM* | *CN* constraint combination (TT *green*, TF *red*, FT *teal*, and FF *blue*). In an effort to better understand our model, we have attempted to characterize how each parameter affects the model's behavior below and in **Table 3**.

For the majority of cases, *CN*=True curves closely followed those with *CN*=False; this is evidenced by the similarity between the FF and FT curves in most plots of **Figures 3** and **4**. However, by plotting the number of infected Agents per quadrant we found that *CN*=True causes a small delay in the spread of infection. **Figure 5B** shows how quadrant 1 has a much higher infection rate for the first several hundred

timesteps when *CN*=True as compared to when *CN*=False in **Figure 5A**. The constraint also introduces more sporadic fluctuations in infection spread as the simulation progresses. Furthermore, imposing both constraints over many time steps masks the effect of *CM*=True. This can be observed in every plot in **Figure 3** except for $p_{stay} = 99.9\%$ since all TT curves reach larger values than TF curves.

Parameter	Effect on Model Behavior Overall	Effect on Model Behavior When Constrained
R_0	Dramatically affects the rate of infection. For instance, when $R_0 = 10$, the graph takes on a steep sigmoidal shape as opposed to a much less severe slope when $R_0 = 1$.	When $R_0 = 1$, the *CM*=True curves shift to the right and the rate of infection plateaus around 3000. When $R_0 = 10$, there is a similar curve shift.
$T_{E\to I}$	When $T_{E\to I}$ is large, the model does not behave differently than the default setting. When $T_{E\to I}$ is small, fewer Agents become infected overall.	*CN*=True alone was insufficient to affect the graph's behavior. *CM*=True decreases the number of infected.
$T_{I\to R}$	A large value slows down the spread of infection dramatically. A small value causes complete saturation very early on, similar to the case when $R_0 = 10$.	When $T_{I\to R}$ is small, the FF curve rises faster than in the default case. In addition, when *CM*=True, the maximal rate of infection is lower despite the peak occurring sooner. However, when $T_{I\to R}$ is large, the system does not appear to reach a maximal infection rate.
p_{stay}	Because p_{stay} is only altered when *CM*=True, it only affects the constrained model.	Using Default *CM*=False as a baseline (when p_{stay} is typically 20%), we see that $p_{stay} = 50\%$ causes slightly more infected cases, $p_{stay} = 90\%$ causes slightly less infected cases, and $p_{stay} = 99.9\%$ effectively stops disease spread.

Table 3. A summary of the effects of changing each parameter on the model.

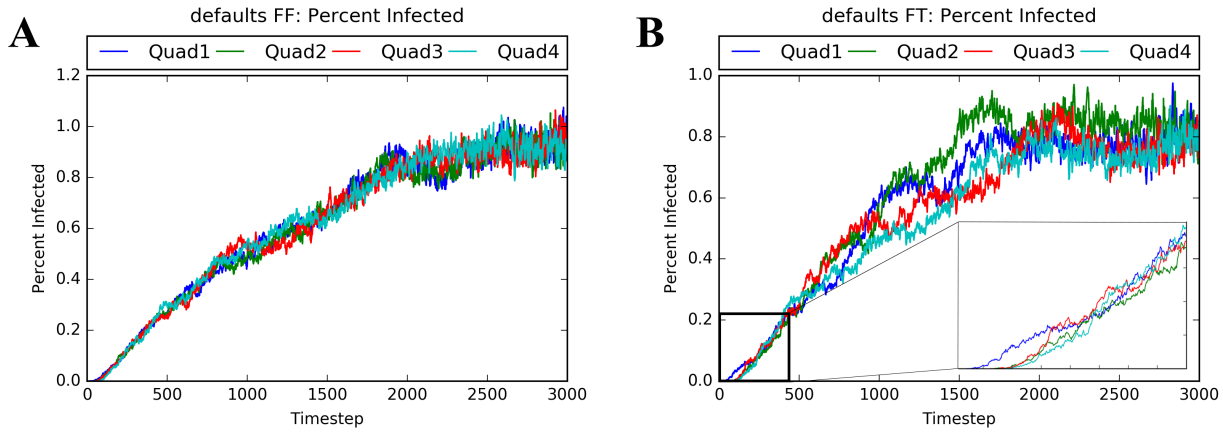

Figure 5. A) Plot of the number of Infected Agents per quadrant using the default parameter settings (*CM*, *CN* = False, False). **B)** Plot when the network is constrained (*CM, CN* =False, True). Each trajectory represents the number of infected agents per timestep separated by quadrant. For instance, Quad1 represents the lower-left quadrant of the constrained graph, where the simulation begins with a single Infectious Agent.

Discussion

Since we had well-defined model parameters at the start of our simulation and reliable sources for their default values[3,11], we had some expectations of what the effect of each would be. For instance, we expected the rate of infection to increase dramatically as R_0 grew larger. This is obviously true since R_0 represents the number of expected secondary cases from a single infectious individual. When we increased R_0 from 1 to ten, we observed this exact behavior. Similarly, the model behaved as expected for $T_{I \to R}$. We expected that decreasing $T_{I \to R}$ would increase the infection rate due to the fact that Infectious Agents must infect the same number of Agents in a shorter amount of time. In contrast, we expected that a large $T_{I \to R}$ would decrease the infection rate. Again, the model exhibited these expected behaviors and seemed to behave reasonably.

We also observed some interesting behaviors that we did not expect. We thought a short $E \to I$ transition time, $T_{E \to I}$, would result in more Agents being infected overall. Our logic was that with a shorter delay between the E and I states, more Agents would become Infected over a simulation run. Instead, we observed that the number of infected Agents grows faster with a *decrease* in $T_{E \to I}$. We believe this can be explained by the way Agents move in our model. A decrease in $T_{E \to I}$ gives Exposed Agents more time to traverse the grid and diffuse amongst populations of Susceptible Agents. In doing so, there are actually more opportunities for an Agent to spread disease once it transitions from $E \to I$ because that Agent has likely moved to a node with more Susceptible Agents.

Overall, the most interesting behavior that we observed was from varying the p_{stay} parameter. We expected that any increase in p_{stay} would slow the spread of disease due to the fact that the Infected Agents would be limited in their ability travel far and infect others. It was true that increasing p_{stay} from 20% to 99.9% was extremely effective at slowing the spread of disease. Additionally, increasing p_{stay} from 20% to 90% was still effective but significantly less effective compared to p_{stay} = 90%. What was most surprising was that a moderate increase in p_{stay} from 20% to 50% made the disease spread *faster*, which goes completely against our intuition. It is unclear why a moderate increase in p_{stay} has the opposite effect compared to a dramatic increase.

To investigate this behavior, we visualized the spread of Ebola by constructing GIF movies for each value of p_{stay} (see GIFs attached). The GIFs depict the 14x14 graph with nodes colored according to the fraction of Agents that are infected at that node; white nodes have no Infected Agents while dark red nodes are completely full of Infected Agents. Shades of pink represent nodes with varying concentrations of Infected Agents. We noticed that in **GIF 1**, when *CM*=False (i.e. $p_{stay} \leq 20\%$), the Infected Agents

diffuse across the graph rapidly. In contrast, **GIF 2** shows how when *CM*=True and p_{stay} = 99.9%, the Infected Agents are never able to travel as far as the top right node. As a result, the simulation never reaches the fully diffuse state and it is obvious that the infection rate is extremely low. **GIF 3** depicts *CM*=True and p_{stay} = 90%. We can observe that although the diffusion of the disease is slower than in **GIF 1**, eventually the model reaches the diffuse state and the behavior is not much different from when *CM*=False. Each of these behaviors are reasonable. Finally, we compared these GIFs to **GIF 4**, the case when *CM*=True and p_{stay} = 50%. To reiterate, we expected that any increase in p_{stay} would slow the spread of disease. However, the movement of Infected individuals in **GIF 4** is hardly distinguishable from those in **GIF 1**, the default case, and warrants further investigation. One possible explanation for this behavior is that *CM*=True and p_{stay} = 50% provide a "perfect storm" of parameters; Agents move slowly enough to create relatively stationary and highly-infected nodes while also moving fast enough to spread the disease to far reaching parts of the graph.

We had hoped that introducing a constrained grid in our model may elicit some underlying behavior in the spread of disease. However, throughout our analysis, we observed that CN appeared to have a negligible effect on the model's behavior. There are at least two possible explanations for this behavior. The first is that the more connected a node is, the less likely an Agent will remain at that node during the next time step. Thus, by connecting the quadrants in the center of the graphs, we did not significantly affect the flow of individuals throughout the nodes. Our model thus provides inconclusive evidence towards the impact of constrained grid on disease spread, and need further work in this area.

Conclusions & Future Work

We built an Agent-based model for Ebola that incorporates a modified SEIR design to study the spread of infection. We performed an initial evaluation of two novel constraints (*Constrained Movement*, and *Constrained Network*) with a variety of different parameter conditions. We were able to conclude that our model was significantly affected by the parameters R_0 and $T_{I \to R}$, resistant to changes just from constraining the network alone, and sensitive to constraining the movement of Infectious Agents. Additionally, when constraining the network as well as the movement of Infectious Agents, the model could recover from the effects of constrained movement after a long period of time.

In the future, our work could be extended in a number of ways. In particular, the effect induced by a constrained network merits more attention. We constructed a simplistic four-quadrant constraint in our model and did not observe any dramatic shifts in behavior as a result. We believe our choice to connect the grid in the center may not have sufficiently limited the spread of disease. This could be addressed by connecting the nodes along the periphery of the grid instead of in the middle. Another possible explanation

is that we did not constrain the graph as significantly as we anticipated, and increasing the number of disconnected regions could produce an observable effect. Both of these situations require further investigation to elucidate the behavior of the model under spatial constraints.

As described in the previous section, the p_{stay} parameter exhibited counterintuitive behaviors that cannot be fully understood through our analysis. This highlights the importance of performing parameter scans to measure the robustness of a model. For each parameter, we tested only low, high, and default values, but we need to look at more intermediate values to better understand the behaviors of each parameter. We propose investigating many more values of p_{stay} to better understand its effect.

We would also like to experiment with the number and location of the Infected Agents that are initialized at the start of the simulation. In every simulation we completed, we initialized a single Infected Agent in the bottom-left corner of the grid. If we want to further answer questions about how the spatial organization of Infected Agents and Susceptible Agents affects disease spread, it could be beneficial to investigate how particular starting points within different grid connectivities affects the model's behavior. For instance, we could instead initialize Infected Agents at random across the graph, or start with more than one. Moreover, it may be advantageous to try different seeds for our random number generator since it may produce slightly different results for the stochastic portions of our simulation.

There is an interesting challenge when comparing the spread of disease as predicted by a model and real-world data. Namely, while the real data is being collected, there are active measures in place to prevent the spread of the disease. In its current state, our model simulates the unimpeded spread of Ebola through a population. It would be interesting to include a more explicit quarantine constraint as well as the ability to designate certain Agents as Immune. By introducing Immune Agents at the start of the simulation, we would be able to estimate the fraction of the population that would need to be immunized in order to minimize the spread of Ebola. Quarantine methods could be applied at different time steps in the simulation to study their effects on disease progression. These types of experiments could help guide policies for future disease outbreaks.

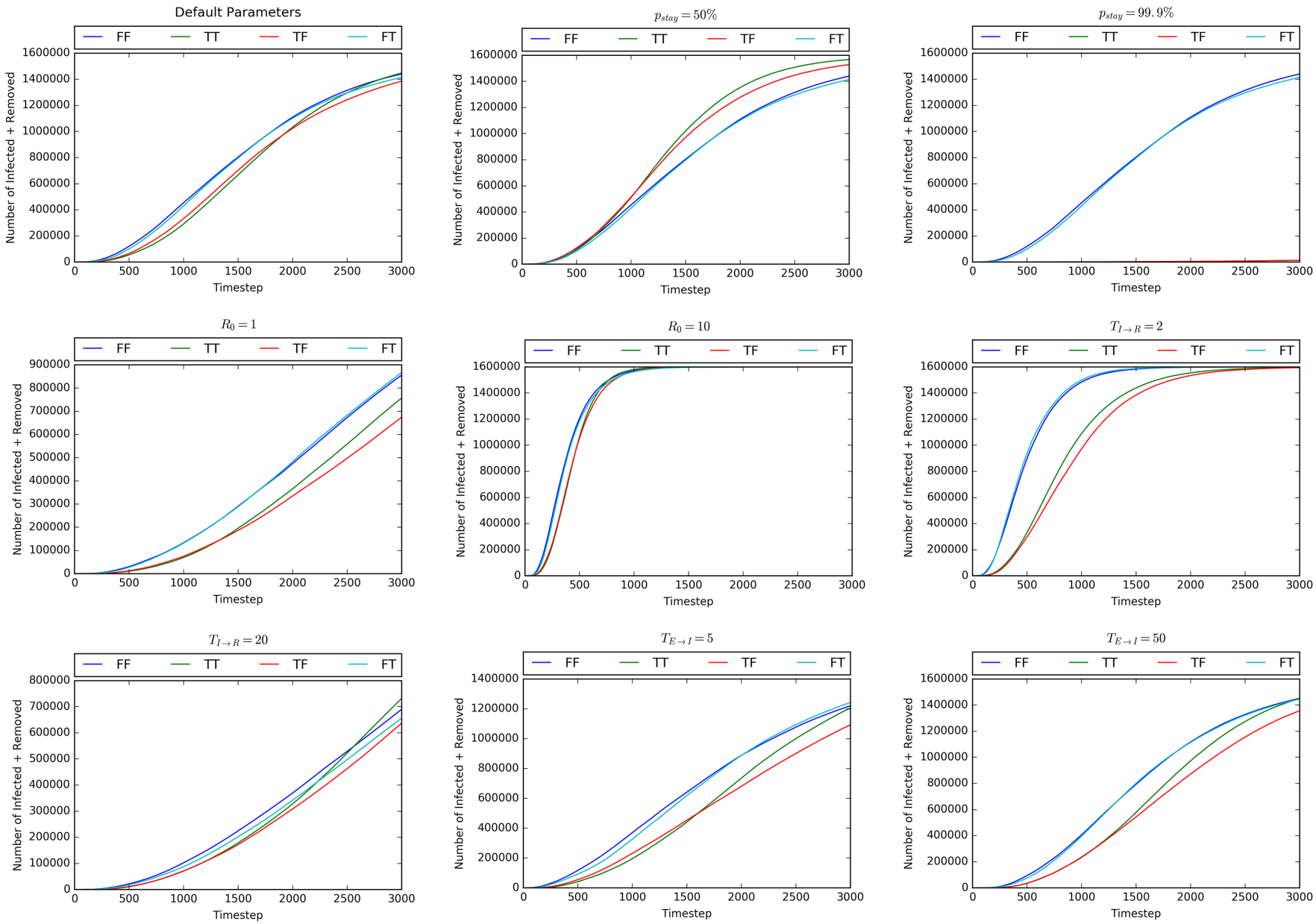

Figure 3. Plot of the number of infected and removed Agents per time step while varying a single parameter. Each plot contains 1 trajectory for each constraint of our model: *None* (FF), *Constrained Movement* (TF), *Constrained Network* (FT), and *Both* (TT). Note that for $R_0 = 1$, $T_{I->R} = 20$, and $T_{E->I} = 5$, the y-bounds have been shortened.

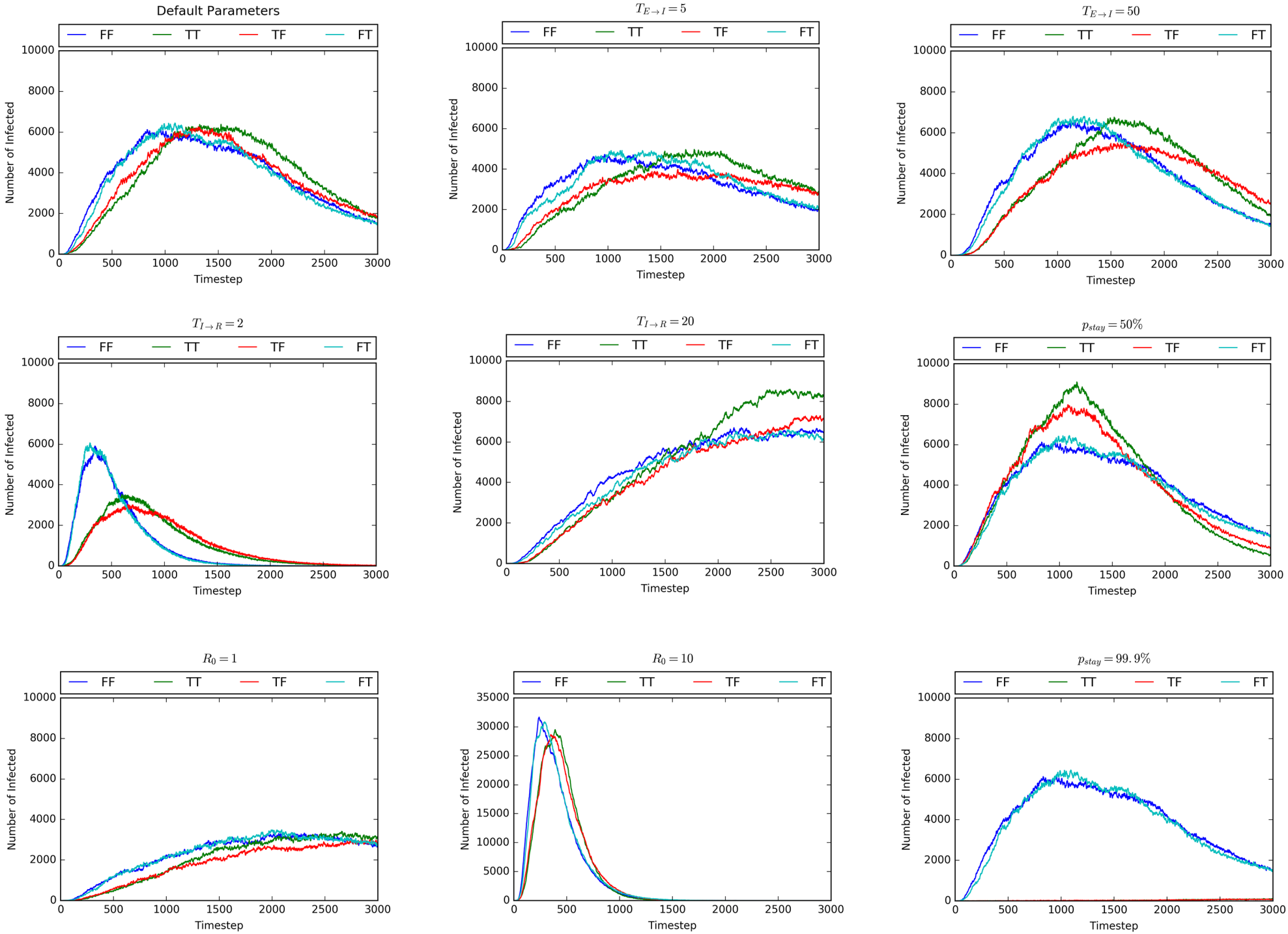

Figure 4. Plots of the number of infected Agents per time step while varying a single parameter. Each plot contains 1 trajectory for each constraint of our model: *None* (FF), *Constrained Movement* (TF), *Constrained Network* (FT), and *Both* (TT). Note that for R_0 = 10, the plot's y-bounds have been extended.

References

1. *The World Health Organization Situation Report for Ebola Virus Disease*. (2016) http://apps.who.int/iris/bitstream/10665/208883/1/ebolasitrep_10Jun2016_eng.pdf?ua=1
2. *WHO publishes list of top emerging diseases likely to cause major epidemics*. WHO. www.who.int/medicines/ebolatreatment/WHO-list-of-top-emerging-diseases/en
3. Nishiura, Hiroshi, and Gerardo Chowell. "Theoretical Perspectives on the Infectiousness of Ebola Virus Disease." Theoretical Biology and Medical Modelling, BioMed Central, 6 Jan. 2015, tbiomed.biomedcentral.com/articles/10.1186/1742-4682-12-1.
4. Chowell, Gerardo, Viboud, Cecile, Hyman, James M., Simonsen, Lone, *The Western Africa Ebola Virus Disease Epidemic Exhibits Both Global Exponential and Local Polynomial Growth Rates*. (2015). PLOS Current Outbreaks. Doi: 10.1371/currents.outbreaks.8b55f4bad99ac5c5db3663e916803261.
5. Morrow, Jim, *Modeling the Spread of Ebola.* (2015) https://sites.math.washington.edu/~morrow/mcm/mcm15/38725paper.pdf
6. Ngwa, Gideon, Teboh-Ewungkem, *A Mathematical Model with Quarantine States for the Dynamics of Ebola Virus Disease in Human Populations.* (2016) Computational and Mathematical Methods in Medicine **2016**:9352725. http://dx.doi.org/10.1155/2016/9352725
7. Kiskowski, Maria, *A Three-Scale Network Model for the Early Growth Dynamics of 2014 West Africa Ebola Epidemic*. (2014) PLOS Currents Outbreaks. Doi: 10.1371/currents.outbreaks.c6efe8274dc55274f05cbcb62bbe6070
8. Bartlett, Jeff, Devinney, James, Pudlowski, Eric. *Mathematical Modeling of the 2014/2015 Ebola Epidemic in West Africa*. SIAM **9:**87-102. https://www.siam.org/students/siuro/vol9/S01380.pdf
9. Martin del Ray, Angel, Sanchez, Gerado. *A CA Model for Mobile Malware Spreading Based on Bluetooth Connections*. (2014) International Joint Conference Advances in Intelligent Systems and Computing 239. Doi: 10.1007/978-3-319-01854-6_63
10. *New Cases per Epi Week for Guinea*. Ebola Data and Statistics, World Health Organization. (2016) http://apps.who.int/gho/data/node.ebola-sitrep.ebola-country-GIN-latest?lang=en.
11. *Ebola Virus Disease Fact Sheet*. Data and Statistics, World Health Organization. (2017) http://www.who.int/mediacentre/factsheets/fs103/en/