Fitting Contact Networks to Epidemic Behavior with an Evolutionary Algorithm

Daniel Ashlock and Elisabeth Shiller

Abstract—Epidemic models often incorporate contact networks along which the disease can be passed. This study incorporates a restarting-recentering evolutionary algorithm, previously developed to locate extremal epidemic networks, together with a new representation, the toggle-delete representation, for evolvable networks. The goal is to locate networks that were likely to have produced a given epidemic behavior. This goal subsumes a new fitness function for driving selection in network evolution. Earlier representations used networks with a fixed sequence of contact numbers. The new representation can add and remove edges from the network, permitting a search that varies contact numbers within the network. A parameter setting study is performed on an epidemic profile obtained from an random network and then tested on a bimodal profile invented by the researchers. The algorithm succeeds in producing networks that cause epidemics run on them to mimic the specified epidemic profiles.

I. INTRODUCTION

WHEN a disease spreads through a social network, the details of the network topology influences the course of the disease in the population. Various network properties, such as the distribution of the number of contacts, are defined so that they create simulated social networks that resemble patterns of contact in the real world. These properties are estimated by real-world surveys of human behavior. Interactions that may result in infection transmission vary depending on the disease being studied and the customs of the society being modeled. These interactions give the connection architecture of a social network and can be based on sexual, social, business, or other behaviors. Survey data might contain the mean number of contacts (degrees of network nodes), distribution of the number of contacts of different individuals, or statistical data on mixing rules between different groups. The different groups can be based on gender, age, race and geographic locations [11], [6], [10]. In this study, we seek to fit plausible networks to a record of the number of individuals that became infected in each time step of the epidemic.

The most common type of information collected in a public health survey is the number of contacts each individual in the sampled population has. The actual topology of the network of contacts is much harder to obtain. That is what motivates this study’s goal of finding networks that are most likely to yield the observed behavior. Obtaining information about the actual topology of a social network requires a much greater invasion of individual privacy and is often inaccurate because of the potential stigma involved in the spread of many diseases. In this study, networks are optimized to obtain a best match to the number of individuals that fall sick in a given time period of an epidemic using a sum-of-squared-error (SSE) metric. We call the best match the epidemic profile match fitness.

This study uses a simple mathematical model of disease spread: the susceptible, infected, removed or SIR model. In its simplest form this model assumes a well-mixed population, but we modify it to permit infection to spread only along network links. The network based model is broadly used in epidemiological research [7]. A pool of individuals is divided into three groups. Those that have not yet contracted the epidemic disease are termed susceptible. Individuals that currently have the epidemic disease are termed infected and are assumed to be able to infect others with the disease. Those that have had the disease in the past but are no longer able to infect others are termed removed. The exact meaning of removed depends on the disease, encompassing states as diverse as permanent immunity and death. In the SIR framework, an individual can have a disease at most once; once removed they stay that way.

An SIR epidemic is initialized with all but a few individuals in the susceptible state and the remainder (in this study, one) placed in the infected state. The epidemic disease is assumed to have a uniform duration equal to the length of a time step in the epidemic. In each time-step, every susceptible individual has a chance $\alpha$ of becoming infected by each adjacent infected individual in the network. The chance of becoming infected along different network links are independent. After probabilities of infection have been evaluated and newly infected individuals identified, those individuals that were previously infected are moved to the removed state.

We now precisely define the objective measure of fitness function used to assess the quality of networks in this study. An epidemic profile is a specification of the number of individuals that get sick in each time period of the epidemic. An example is shown in Figure 1. The fitness function incorporates information from $n = 50$ simulated epidemics, using the network node with the lowest index number as patient zero. Patient zero is the individual that starts in the infected state. The sum squared error of the number of individuals that fall sick in a best match to the number of individuals that fall sick in a given time period of an epidemic using a sum-of-squared-error (SSE) metric. We call the best match the epidemic profile match fitness.
sorted into increasing order

\[ E_1 \leq E_2 \leq \cdots \leq E_n \]

and the fitness of a network \( \mathcal{N} \) is:

\[ fit(\mathcal{N}) = \sum_{k=1}^{n} \frac{E_k}{k} \quad (1) \]

This is a linearly weighted sum of SSE values, a choice that requires some explanation. The reason for the weight sum is amelioration of the problem that a given epidemic may be a very rare event even on the network that is most likely to produce it. Sorting the SSE values means that the sampled epidemics that did the best job of approximating an epidemic profile are the one to which the fitness function pays the most attention. If we were testing the quality of the network as a final result this would be an unsupported bias; a fitness function, however, needs to create a fitness gradient toward the desired result. If an epidemic profile is a common result on a network then this fitness function will return a good value. If, on the other hand, an epidemic is a rare event on a network then the fitness function will return a good value only if the rarity is an intrinsic property of the epidemic profile.

The fitness function is not used as an absolute measure of quality, rather it is a relative measure of the suitability applied to pairs of networks. This fitness function is therefore likely to enhance the location of more likely networks for a given profile. The problem of locating a particular network that yields a given profile is not well posed; there is no reason to think there is a unique best network for a given profile, not even if the profile was produced by a network. The goal is thus not to find the “correct” network for a given profile, but to find a network that was likely to produce it that can be used in intervention simulation and other forms of study. We note that the fitness function is based on random samples and is hence stochastic. This is a factor that must be managed in the design of experiments.

The remainder of this study is structured as follows. Section II specifies the novel evolutionary algorithm used here. Section III gives the design of the experiments. Section IV gives and discusses the results of the experiments. Section V draws conclusions and outlines possible next steps.

II. Evolutionary Network Search

This paper follows on earlier studies [1], [2], [3], [4]. The first of these used evolutionary computation to search for extremal networks in which each member of the population had exactly three contacts. Several different representations for networks were tested and it was found that different representations search distinct parts of network space. They also yield substantially different ranges of fitness values. In all subsequent studies, including this one, the network representation was changed to the one described below. This solved many problems caused by the overly-restrictive representation in the first paper. The representation common to the later papers enables evolutionary search that is general across all networks that share the same degree sequence. The degree of a node in a contact network is the number of its neighbors. The degree of a node is also called the contact number. The study [1] introduced the edge-swap representation while [3] characterized the behavior of this representation when maximizing epidemic lengths on networks with a variety of different degree sequences. The study in [4] introduced an additional fitness function for driving network evolution. It used a tool called diffusion characters[5] to assess how different and variable the social networks were when evolved with different fitness functions.

Perhaps the most critical feature of an evolutionary algorithm is the representation used. The representation encompasses both the data structure used to store problem solutions in the computer and the variation operators that act on it. For evolutionary search to be effective, it is beneficial if crossover operators are able to mix-and-match features from different solutions so that they yield superior solutions. Likewise, mutation operators should be able to make small or local changes in structures so that good structures can be made incrementally better. If variation operators do not exhibit these features, which amount to ensuring heritability of solution quality from parent to child, then an algorithm that simply generates random problem solutions, checks them, and then saves the best may well outperform the evolutionary algorithm. Evolutionary computation algorithms must also have a technique for initializing their populations. The most common method is to sample them from an ad-hoc distribution on the space of potential solutions.

A. The Evolvable Network Representation

The distribution of degrees for contact networks are much studied. For example, contact networks for sexually transmitted diseases are typically not very regular [8], [9] and often contain a small number of nodes with a very high contact number. The correct statistical model of the distribution of numbers of contacts is still an object of research. In this
study evolution is used to attempt to discover the correct distribution of contact numbers.

The representation used is a string of large integers that are interpreted in triples to yield a series of commands for modifying a graph. The object that is evolving is thus a generative representation that specifies how to modify a graph. For a network with $N$ nodes a triple $a_1, a_2, a_3$ is interpreted as follows:

| $a_1 \mod N$ | index of node to operate upon.
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>$a_2 \mod 2$</td>
<td>remove neighbor (0) or toggle neighbor (1).</td>
</tr>
<tr>
<td>$a_3 \mod k$</td>
<td>$k$ is either the number of neighbors or number of nodes in the graph, depending on whether $a_2 \mod 2$ is 0 or 1 respectively. This entry selects the second node to act on.</td>
</tr>
</tbody>
</table>

A triple selects a node of the network and either (a) removes the edge making it adjacent to one of its neighbors or (b) picks another node in the graph at random and toggles the edge between it and the selected node. A toggle operation changes the status of the pair between connected or not-connected, becoming what it was not. Section III gives the details about the number of triples used in each experiment.

The contact network of a population is typically sparse (far fewer nodes are neighbors of an average node than are non-neighbors). This generative representation favors sparse graphs since one of the two commands removes edges while the other toggles them. In a sparse graph a “toggle edge” command is almost always yields an “add edge” result. This makes the generative representation, more-or-less, contact number neutral. This is the correct design for an evolution driven search.

Since this representation only yields directions about how to modify an existing network, a starting network is also required. This network is specified as a part of the design of experiments in Section III. We call this representation the toggle-delete representation for network evolution.

B. The Recentering-Restarting Algorithm

If a very large number of edges are toggled then almost any network may result. The toggle-delete representation has a similar character, except that it favors sparse networks. All networks are still possible but the probability measure is shifted in favor of graphs with relatively few edges. When a space is constrained so that direct encoding of a uniform distribution is difficult or impossible, the distribution of objects at the end of a long random walk resemble a uniform distribution. This means that a very long chromosome in the toggle-delete representation could equitably search the space of networks for one that was good at exhibiting the given epidemic profile. This would, however, be a representation with an exceedingly high dimension. A small number of toggles and edge-deletions do not modify the network much, leaving little for evolution to do. Earlier studies [2] showed that a representation long enough to yield effective search for better networks is exceedingly slow. To address this problem the recentering-restarting evolutionary algorithm was applied.

The normal role of restarting in an evolutionary algorithm is to conserve machine time. When a heuristic measure of convergence or progress detects that the algorithm is unlikely to make additional progress a new population is generated and evolution begins again. Aside from the new information in the initial populations there is no difference in the starting point of the new instance of evolution. The initial network that is modified by the toggle-delete representation yields an opportunity: the population’s current best network can be used to replace the central network when the algorithm restarts. This is called a recentering restart with the initial network playing the role of the “center” of the search. This technique has a number of advantages. Search through the space can proceed in a sequence of steps, each using a toggle-delete representation of moderate length. This allows the eventual location, after several recentering restarts, of networks at a distance from the initial network that would otherwise require a very long (and inefficient) string of toggle-delete operations. In addition, recentering permits the algorithm to retain some part of the gains made by the previous evolutionary effort.

At this point the reader may wonder how retaining the results of previous evolution could be an advantage given that the usual purpose of restarting is to acknowledge that evolution is stuck and should start over. Notice that the toggle-delete representation makes a series of edits to the starting network and so the starting network is exceedingly unlikely to be present in the population - this would require a sequence of edge modifications that exactly add and remove the same edges. When the algorithm recenters, it loses whatever networks were present in the population and now searches a different part of the space. The algorithm is centered on, but searching at some distance from, the network.
that replaced the previous starting network. This notion of distance can be made precise.

**Definition 1:** The *edge-toggle* distance (ET-distance) between two networks is the minimum number of edges that must be deleted or added to transform one network into the other.

The edge modifications appearing in an instance of the toggle-delete representation probably do not form a witness to the ET-distance between the center and the network resulting from application of the instance. In other words, the members of the population do not lie on a precise circle relative to the edge-toggle metric. The distribution of distances, however, has a large mode just short of a radius, in the ET-distance, to the number of toggles and deletes used. This means that when the algorithm centers search takes place in a fairly narrow spherical shell with the starting network at its center. Figure 2 gives a diagram suggesting how recentering-restarting evolution progresses through the space.

When the algorithm recenters, fitness gets substantially worse. Over many recenterings it is common for the fitness to get better, as shown in Figure 3, but it is not invariably the case. In order to keep track, the current best graph used in recentering is saved before each recentering takes place. These graphs are then all compared to one another with 500 simulated epidemics, rather than 50 and then the best of them is reported as the result of the algorithm. The use of 500 simulated epidemics decreases the stochasticity of the fitness function and permits more accurate comparison.

### III. DESIGN OF EXPERIMENTS

The representation used is an array of integers whose interpretation was given in Section II. These arrays are initialized to integers in the range $0 \leq n < 10,000$. The range is large enough to be almost uniform when modular arithmetic is used in the interpretation. The evolutionary algorithm in question uses size seven single tournament selection on a population whose size is varied in different experiments. This is a steady state algorithm with time measured in *mating events* so that different population sizes are still given the same number of mating events when
other algorithm parameters are held constant. A mating event consists of a single instance of tournament selection, generation of children, and placement of those children back into the population.

The variation operators used are two-point crossover and point mutation. A new individual undergoes 1-3 point mutations, with the number of mutations selected uniformly at random. A point mutation generates a new integer in the range \(0 \leq n < 10,000\) and uses it to replace one of the integers at a position selected uniformly at random.

The simulated epidemics used to compute the fitness function adopt a value of \(\alpha = 0.5\). The fitness function is stochastic, so whenever two networks are chosen to reproduce, the fitness of the children is evaluated and the fitness of the parents is re-evaluated. If this is not done then a fitness evaluation that is an outlier can grant a parent excessive reproductive rights.

A parameter setting study was performed for the unimodal epidemic profile shown in Figure 1. Each experiment consists of 30 replicates of the evolutionary algorithm using a population size of 60 with different random number seeds. The first set of experiments compared gene lengths of 180, 240, and 300 using 10, 20, 30, 40, 50, or 60 epochs of length 10,000. An epoch is the number of mating events before a recentering takes place. This is not a resource-neutral comparison; it is intended to see if the algorithm can benefit from more evolutionary time for each given gene length.

A second set of experiments was performed comparing population sizes of 10, 60, 120, 240, and 480 for 40 epochs of length 1000 and 2500. The short epoch lengths permit more rapid comparison of population sizes. A third set of experiments was performed comparing epoch lengths of 1000, 2500, 5000, 7500, and 10000 with population size 60, gene length 120, and 40 epochs. This experiment is not resource neutral, and again we are testing to see if added resources are valuable. A fourth set of experiments compares gene lengths of 60, 90, 120, and 150 with 40 epochs of length 2500 and a population size 60.

Once the parameter setting study had been performed with the fitness function using the epidemic profile given in Figure 1, an experiment was performed with the more difficult profile given in Figure 4. This latter experiment used the values found to be best in the parameter setting experiment, given in Section IV.

In all the experiments the mean best fitness over 30 replicates of the experiment was used as the statistic for comparison. This was done because the relationship between mean and best performance has proven difficult to characterize in this system. The best performance is the statistic of interest until the theory of this type of system can be understood more completely.

It remains to specify the starting graph. All experiments used networks with 128 nodes. This is relatively small for a population in an epidemic model, but small enough to permit the algorithm to run quickly enough to perform a large parameter study. A random graph consisting of nodes with approximate degree 4 was used. This is generated by adding edges at random that do not result in network nodes with more than four members. Typically 1 to 2 nodes with 2 or 3 neighbors result with all others having four neighbors.

IV. RESULTS AND DISCUSSION

Figure 3 shows the average population fitness over the course of evolution. Fitness is sampled once every 2,000 mating events. The recenterings of the algorithm are clearly visible in the fitness trace: a large upward spike after each recentering. It is important to note that the general trend of the fitness is downward. Recentering-restarting evolutionary algorithms were introduced in [3]. It was found that even though restarting causes a substantial worsening of fitness values, failure to restart leads to a lack of progress.

The best collection of algorithm settings found in the parameter study was gene length 120 (yielding 40 toggle-delete commands), population size 60, epoch length 10,000 using 60 epochs. The superiority of the longest epoch length is consistent with the results in the earlier studies [3], [4] on evolving extremal epidemic networks.

The results of varying the number of epochs are shown in Figure 6. Giving the algorithm more epochs is found to be highly significant for all three gene lengths tested. Varying population size yields the outcome shown in Figure 7. These results are also found to be highly significant.

![Figure 6](image_url)

Fig. 6. Holding population size constant at 60, average best fitness is compared for three different gene lengths. \(\chi^2\) tests yield significance of \(p < 1 \times 10^{-6}\) for gene length 180, \(p = 0.000291\) for gene length 240, and \(p = 0.001265\) for gene length 300.

Figure 8 gives the results of varying epoch lengths. The results across the entire collection of experiments is not significant, but the shortest and longest epoch lengths are found to be significantly different. This suggests that while giving the algorithm more resources is beneficial, the marginal return on such investment is small. This follows from the fact that very large changes in the resource allocation are needed to obtain a significant benefit.

Varying the length of gene used, shown in Figure 9, did not yield significant differences. This is at odds with the results
in [3] which found this to be an important parameter in the context of a restarting-recentering evolutionary algorithms. The fitness function and representation used in this study are different from the earlier study. This suggests that the fitness landscape in the current study is more amenable to incremental search making the size of the jumps made during recentering less critical.

The following question arises. How effective are the evolved networks at exhibiting the desired epidemic profile? Figure 5 shows the mean behavior of 100 simulated epidemics on the best network evolved with the best parameter set. The average epidemic behavior closely tracks the desired profile.

A. The Experiment using a Bimodal Profile

The unimodal epidemic profile shown in Figure 1 was generated by running an epidemic on a randomly generated network. This means it is obvious that it was possible to find networks that exhibited this behavior. This was done intentionally, to create a relatively soft target for performing parameter studies that characterize algorithm behavior. The epidemic profile given in Figure 4 was simply picked arbitrarily to be bimodal and so there was no a priori reason to believe that it would be easy, or even possible, to find a network that yielded this approximate behavior.

Figure 10 shows the fitness trace of the second epidemic profile juxtaposed with confidence intervals on the mean of 100 simulated epidemics. The traces of those epidemics are in grey in the background. The fitness trace (in the upper panel of Figure 10) shows that the optimizer is making progress in optimizing the network. The average of the 100 simulated epidemics shown in the lower panel of Figure 10, is very close to the specified bimodal profile. This clearly demonstrates a nontrivial ability to fit networks to epidemic behaviors. Unlike Figure 5 (the analogous figure for the unimodal epidemic profile), this figure was drawn from a single experiment consisting of 30 runs of the evolutionary algorithm. Figure 5 was the best graph from a parameter setting study in which 55 experiments (1,650 runs of the evolutionary algorithm) were performed. This suggests that the parameter tuning was successful and generalized correctly, at least in the case of the bimodal epidemic profile.

Comparing the upper panel of Figure 10 with Figure 3, the reader will note that the initial fitness values at
Fig. 7. Comparing population sizes for two epoch lengths with gene length held constant. A $\chi^2$ test finds differences between population sizes to be significant with $p < 1E^{-6}$.

Fig. 8. Shown is the variation in average best fitness with gene length 120, population size 60, and 40 epochs of varying length. A $\chi^2$ test for the entire data set yields a $p = 0.055$ value while the difference between 1,000 and 10,000 epochs is significant with $p < 1E^{-6}$.

much better for the unimodal epidemic profile. The bimodal problem’s initial population started with fitness values near 250 while the unimodal population starts near 200. They finish near to 100 with the unimodal problem a little lower than the bimodal problem. This strongly suggests that the optimization problem for the bimodal epidemic profile is more difficult.

V. CONCLUSIONS AND NEXT STEPS

Figures 5 and the lower panel of Figure 10 demonstrate that the system described in this study is able to fit a network to a specified epidemic behavior. This study thus stands as a proof of concept for this objective. Verifying and checking the many choices made in the design of both representations and the algorithm are an early priority for future research. The system presented here was constructed largely in analogy with the successful system from [4] that performed a related, but distinct task. It may well be that algorithm performance can be substantially enhanced by systematic re-examination of those choices.

Novel features of the study include searching the space of networks for examples that are likely to generate a given epidemic profile and the introduction of the toggle-delete representation for network evolution. Earlier studies in this series share with the toggle-delete representation the character of being generative representations that evolve a series of modifications of a starting network. Many of the other network editing operators (edge-swap, simplexification, etc.) from the earlier studies could easily be combined with the toggle and delete operation to create richer sets of generative operations for graph evolution. A study to determine which sets of operations are appropriate for each possible network search problem is another potentially rich source of improvements in algorithm performance.

This study confirms the utility of the restarting-recentering evolutionary algorithm for searching spaces of networks. Earlier studies using this type of evolutionary algorithm [3], [4] used both a different representation and a different fitness function. In particular, the earlier studies preserved the number of edges in the network while this study varies them. A comparison with a standard algorithm using a much longer toggle-delete chromosome was not performed because past experience suggested it would yield substantially inferior results; an extension of this work should check this hypothesis.

The networks used in this study all had 128 nodes. This was a compromise of speed (particularly for the parameter study) with a number of nodes large enough to exhibit interesting behaviors such as the bimodal epidemic. The algorithm is fairly fast, 20-30 minutes per run of the evolutionary algorithm, and it will scale linearly with the number of nodes. The current size of networks is on the low end of those useful for epidemiological studies. The technique
Fig. 10. The upper panel shows fitness over the course of evolution in one run of the evolutionary algorithm. The lower panel shows 95% confidence intervals for the mean number of infected in 100 replicates of an epidemic run on the most fit graph for population size 60, gene length 120, and 40 epochs of length 10,000 for the bimodal epidemic profile. The desired epidemic profile is shown without error bars and the tracks of the individual epidemics are shown in the background.

must be scaled to at least thousands of nodes. This makes it critical to develop ideas for speeding up the algorithm.

The two epidemic profiles used in this study demonstrate that the system can fit networks to epidemic profiles, but there are many other possible profiles. A followup to this study should check which profiles are accessible to the system. A profile with four modes, for example, would probably be impossible for networks with 128 nodes simply because there are not enough people in the population of susceptibles to support the epidemic through three low points. In general, more complex profiles will require larger numbers of nodes or, equivalently, populations of susceptibles. The relationship between the complexity of a profile and the size of the susceptible population needed to support it is a potentially interesting area of research.

In this study, the probability of disease transmission within the SIR model is set to 0.5 so as to maximize the variability of epidemic behavior, but no study on this parameter was done. It is possible to vary this parameter or even to assign individual network nodes different transmission probabilities to model different circumstances or individual behaviors. This moves this technology toward evolving agent-based models in network form. Disease spread can be enhanced or held back by environmental factors such as building ventilation or hand washing compliance. In addition to adding a higher degree of realism, having variable disease transmission probabilities might produce networks that model observed epidemics with higher fidelity.

The SIR model was chosen for this study because of its simplicity, not its plausibility. It is not difficult to built fitness functions based on variations of the SIR model. These include SIRS (susceptible infectious recovered susceptible), SEIR (susceptible exposed infectious recovered) and SEIRS (susceptible exposed infectious recovered susceptible) models. A good direction for future research includes investigating these, more complex, models of disease transmission. Finally, stratifying the nodes in male and female, or by age or by susceptibility category potentially permits much higher fidelity of modeling.

REFERENCES


