Comparison of Evolved Epidemic Networks with Diffusion Characters

Daniel Ashlock
Mathematics and Statistics
University of Guelph, Guelph
Ontario Canada N1G 2W1
dashlock@uoguelph.ca

Elisabeth Shiller
Mathematics and Statistics
University of Guelph, Guelph
Ontario Canada N1G 2W1
eshiller@uoguelph.ca

Colin Lee
Mathematics and Statistics
University of Guelph, Guelph
Ontario Canada N1G 2W1
clee04@uoguelph.ca

Abstract—**Epidemic models often incorporate contact networks along which the disease can be passed. This study uses a recentering-restarting evolutionary algorithm to locate likely epidemic networks for six different epidemic profiles containing early peaks, late peaks, and multiple peaks in the number of infected individuals. This study demonstrates that the algorithm can fit a broad variety of epidemic profiles. The difficulty of finding a network likely to produce a given epidemic profile varies between profiles, but all six profiles are fitted well in at least some of the evolutionary runs. A pseudometric on pairs of networks based on diffusion characters is used to assess the networks distribution in the space of networks. Both the scatter of networks evolved to match a single epidemic profile and the between-profile distances are evaluated. The diffusion character based pseudometric separates the networks for some pairs of profiles neatly while others apparently overlap to some degree.**

I. Introduction

When a disease spreads through a social network, the details of the network topology influence the course of the disease in the population. Various network properties, like the number of contacts per person, are used to create simulated social networks that resemble patterns of contact in the real world. Interactions that may result in the transmission of infection vary depending on the disease being studied and the customs of the society being modeled. These interactions induce the connection architecture of a social network and can be based on sexual, social, business, or other behaviors. Survey data may contain the mean number of contacts (mean 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, or geographic locations [1], [2], [3].

This study builds on an earlier study [4] which performed extensive parameter studies on a system that fits plausible networks to a record of the number of individuals that became infected in each time step of the epidemic. We call such a record an *epidemic profile*. In this study, the system is tested on six diverse epidemic profiles and the results are analyzed with a unique tool based on diffusion characters [5], [6].

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. 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. That problem is what motivates the goal of finding networks that are most likely to yield an observed epidemic.

In this study an evolutionary algorithm is used to optimize networks to obtain a best match to an epidemic profile. The figure of merit used is a modified sum-of-squared-error (SSE) metric of simulated epidemics on a network with an epidemic profile. We call this *epidemic profile match fitness* and note that this fitness is minimized to find high quality networks.

This study uses the susceptible, infected, removed or *SIR* model of an epidemic. In its simplest form this model assumes a well-mixed population. In this study it is modified to permit infection to spread only along network links. The network based version of the SIR model is broadly used in epidemiological research [7]. In a SIR model individuals are 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; removal is a durable state.

An SIR epidemic is initialized with all but a few individuals in the susceptible state, a few (in this study, one) placed in the infected state, and no removed individuals. In each time-step of the model every susceptible individual has a chance α of becoming infected by each adjacent infected individual in the network. This parameter is set to $\alpha = 0.5$ in all experiments. The chances of becoming infected along different network links are independent. After probabilities of infection have been evaluated and newly infected individuals

The authors thank the Canadian National Science and Engineering Research Council, the Ontario Council of Graduate Studies and the University of Guelph Department of Mathematics and Statistics for their support of this research.

identified, those individuals that were previously infected are moved to the removed state. The epidemic disease is assumed to last for one time step in each individual that contracts it.

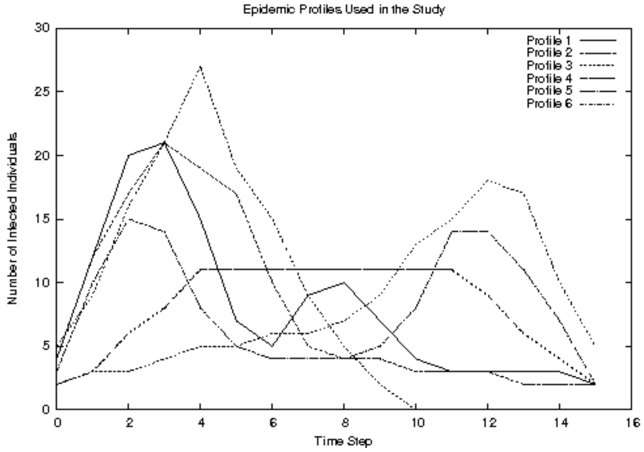


Fig. 1. Shown are the six epidemic profiles used in this study. These profiles are shown individually in Figure 4.

We now precisely define the objective measure of fitness used to assess the quality of networks. The starting point is the epidemic profile. The profiles used in this study are shown in figure 1. The fitness function uses $n = 50$ simulated epidemics, always infecting the network node of index zero. We call this node *patient zero*. The SSE of the number of individuals that get sick in each time step is assessed in each simulated epidemic. These SSE measurements are then sorted into increasing order

$$E_1 \leq E_2 \leq \cdots \leq E_n$$

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

$$fit(\mathcal{N}) = \sum_{k=1}^{n} \frac{E_k}{k} \qquad (1)$$

This fitness function is designed with the following fact in mind: a particular epidemic is unlikely to generate the most likely network profile for the network it actually happened on. The linearly down-weighted sum of epidemics rewards having *some* of the epidemics matching the profile while penalizing epidemics far from the profile relatively lightly. This creates a fitness gradient in the direction of networks that are likely to generate the given epidemic profile.

Equation (1) is not used as an absolute measure of the quality of a network. Rather, it is a relative measure of quality used to compare pairs of networks. This fitness function is therefore likely to enhance the location of networks that have a higher probability of generating a given profile. Note that the fitness function is based on random samples and hence is 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 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.

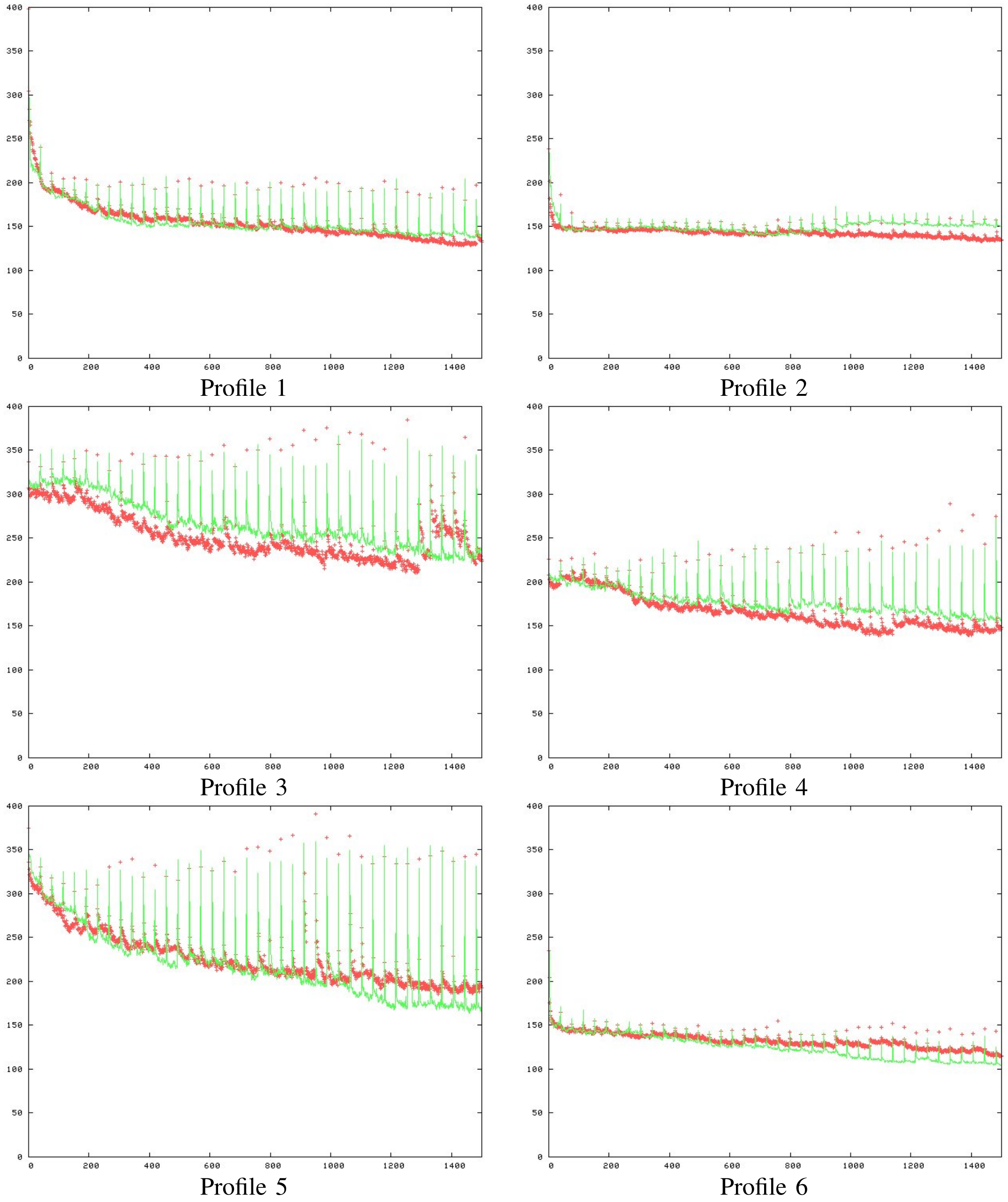
II. Evolutionary Network Search

This paper follows on earlier studies using evolution to create epidemic networks [8], [9], [10], [5], [4]. The first of these used evolutionary computation to search for extremal networks in which each member of the population had exactly three contacts. These networks were optimized to either maximize the number of time steps in an epidemic or to maximize the number of individuals infected in any one time step. Various representations for networks were tested and it was found that different representations search distinct parts of network space. Different representations also yield substantially different ranges of fitness values. In the second, third, and fourth of these studies, the representation was changed to a generative representation. This generative representation is called the *edge-swap* representation. It was used to evolved modifications to an initial graph in a stepwise fashion. This generative representation enabled evolutionary search that is general across all networks that share the same distribution of contact numbers. The study [8] introduced the edge-swap representation, while [10] characterized the behavior of this representation when maximizing epidemic lengths on networks with a variety of different degree sequences. The study in [5] introduced an additional fitness function for driving network evolution. It used a tool called *diffusion characters* [6] to assess how different and variable the social networks were when evolved with different fitness functions. In [4] a new generative representation (also used in this study) was introduced. This study generalizes [4] by applying the new representation to six new epidemic profiles and applying the best of the diffusion character based pseudometrics from [5] to comparison of the evolved networks.

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, crossover operators must be able to mix-and-match features from different solutions so that they yield superior solutions with some positive probability. Likewise, mutation operators should be able to make small or local changes in structures so that good structures are sometimes made incrementally better. These features amount to permitting heritability of solution quality from parent to child. Without these features, evolutionary computation degenerates to generate-and-test. Evolutionary computation algorithms also require a technique for population initialization. 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 that arise from sexually transmitted diseases are much studied. Typically they are not very regular [11], [12], and the correct



Profile 1

Profile 2

Profile 3

Profile 4

Profile 5

Profile 6

Fig. 2. Plots of fitness over the course of evolution for the first evolutionary run of each epidemic profile. Red crosses mark best population fitness while the green lines track average population fitness. The best fitness of the initial population after each recentering is computed as well, yielding the red crosses not below the green lines at the beginning of each recentering. Statistics are taken every 200 mating events.

statistical model of the distribution of numbers of contacts is still an object of research. It is clear that real-world epidemic networks have substantial variation in their contact numbers. In this study, evolution implicitly attempts 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 network. The object that is evolving is thus a generative representation that specifies how to modify a network. 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.**
$a_2 \ mod \ 2$	**remove neighbour (0) or toggle neighbour (1).**
$a_3 \ mod \ k$	**k is either the number of neighbours 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.**

A triple selects a node of the network and either (a) removes the edge making it adjacent to one of its neighbours 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. Section III gives the details about the number of triples used in the experiments.

The contact network of a population is typically sparse (far fewer nodes are neighbours of an average node than are non-neighbours). Uniformly high contact numbers, in effect, simulate a well mixed population. The 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 almost always yields an "add edge" result. This makes the generative representation roughly contact number neutral. The fitness gradient within evolution can then drive the otherwise fairly neutral modification of contact numbers toward fitter regions of the space via selection.

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.

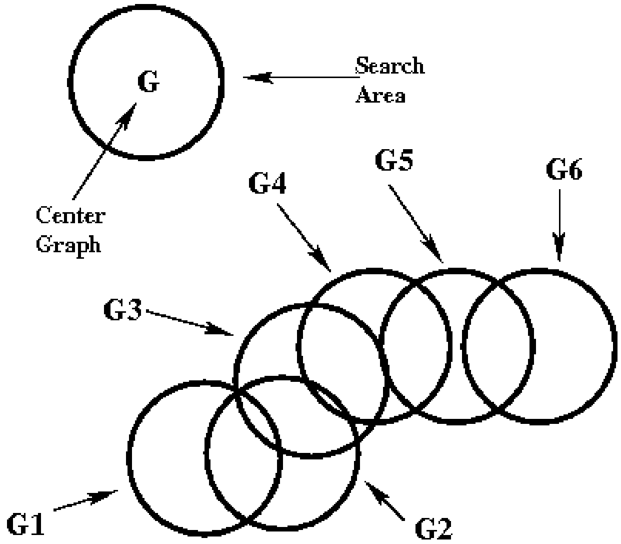


Fig. 3. The progress of recentering-restart through the search space of possible networks (top) and the population structure of a single evolutionary search (bottom, left). G_1 is the initial network, G_2 the first recentering, G_3 is the second recentering, and so on.

B. The Recentering-Restarting Algorithm

If a very large number of edges are toggled then almost any network may result; in essence a large number of edge toggles generates a near-uniform distribution on the space of networks. The toggle-delete representation has a similar character, except that it is biased toward sparse networks. All networks are still possible but the probability measure is shifted to favor sparseness. 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 (in the tens of thousands). A small number of toggles and edge-deletions do not modify the network much, leaving little for evolution to do. Earlier studies [9] showed that a representation long enough to yield effective search for better networks is extremely slow. To address this problem, a *recentering-restarting evolutionary algorithm* was applied.

Restarting an evolutionary algorithm is typically done to conserve machine time. It does this by stopping runs that a heuristic suggests are performing badly or have effectively converged. 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 starting 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 that are at a large distance from the initial network. Recentering also 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 changes to the starting network and so the starting network is exceedingly unlikely to be present in the population of modified networks. This event requires a sequence of edge modifications that add and remove exactly the same edges which is astronomically unlikely. 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. Making a geometric analogy, the members of the population lie near a sphere about the central network whose radius is defined by the number of edge modifications currently being used by the algorithm. When the algorithm recenters, search takes place in a fairly narrow spherical shell with the starting network at its center. Figure 3 gives a diagram suggesting how recentering-restarting evolution progresses through the space.

When the algorithm recenters, fitness can become much worse. Over many recenterings it is common for the fitness to get better. Figure 2 shows the progress of fitness over evolution for a typical run for each of the six epidemic profiles used. The spikes in the fitness plot correspond to recenterings. Since some recenterings yield a loss in overall fitness even after evolution has run up until the next recentering, the graphs used as centers are all saved. The saved graphs are then all compared to one another with 500 simulated epidemics (rather than the 50 used in standard fitness evaluation) and then the best of these is reported as the algorithm's result. The additional simulated epidemics used in this final comparison reduce the stochasticity of the fitness function.

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 to extract the edge modification moves. The representation is sized to have 40 edge moves specified by each population member. The evolutionary algorithm in question uses size seven single tournament selection on a population of size 60. This is a steady state algorithm [13] with time measured in *mating events*. A mating event consists of a single instance of tournament selection, generation of children, and placement of those children back into the population. Evolutionary algorithms in this study are run for 40 recenterings with 7,500 mating events between recenterings. Summary statistics about fitness are saved every 200 mating events. The parameters used are drawn from the best values found in [4], where an extensive parameter study was conducted. Thirty independent replicates of the evolutionary algorithm were performed for each epidemic profile.

The variation operators used are two-point crossover and point mutation. A new individual is subjected to 1 to 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 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 are re-evaluated. If this is not done then a fitness evaluation that is an outlier can grant a parent excessive reproductive rights.

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 was chosen because it is small enough to permit the algorithm to run quickly. 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 neighbours result with all others having four neighbours. We now describe the tools used to analyze the evolved networks.

A. Diffusion Characters

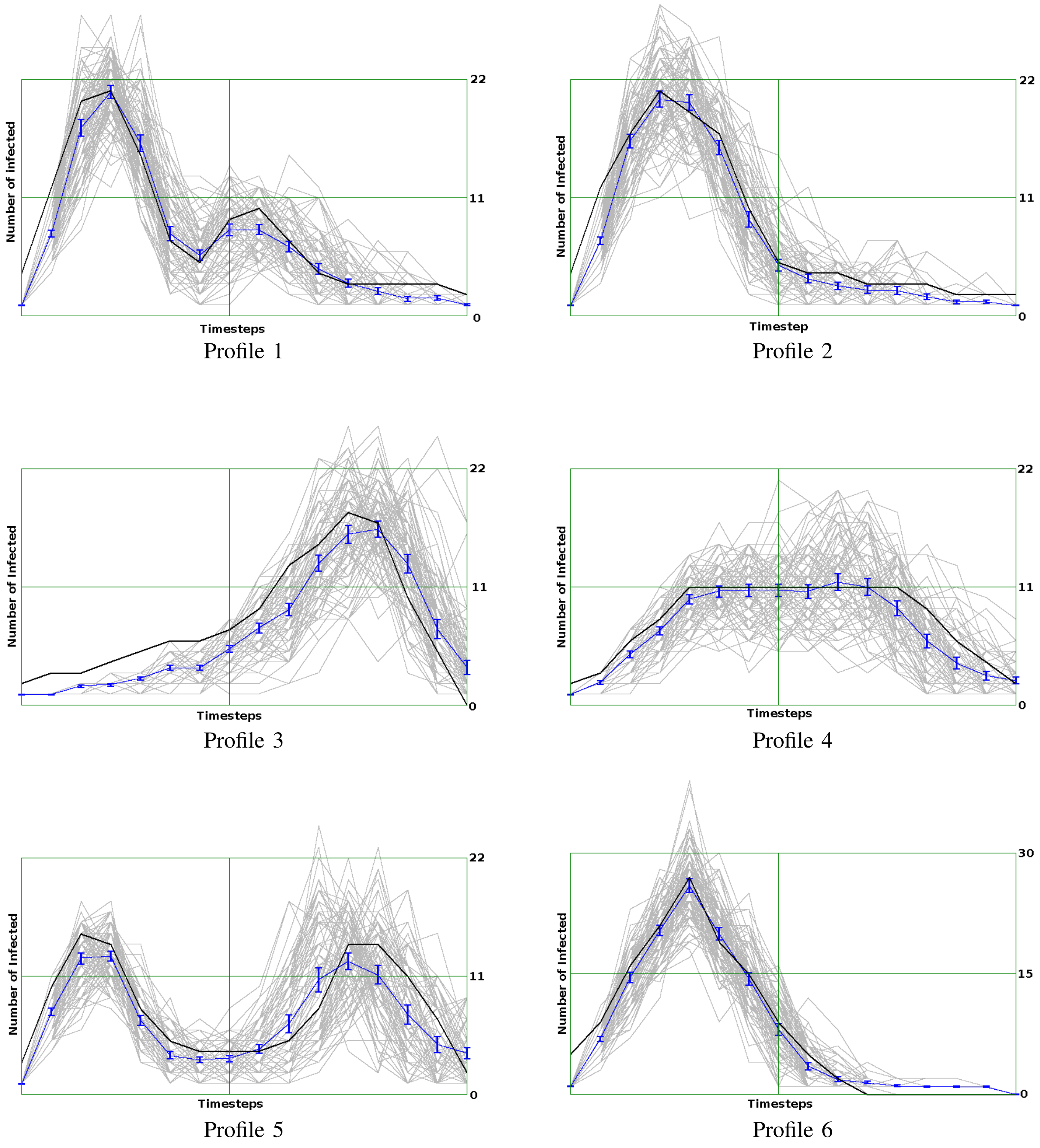
This study applies a technique for comparing the similarity of or approximate difference between networks based on *diffusion characters*. In [5] four different distance measures (pseudometrics) on spaces of networks were derived from the diffusion character matrices of the networks. In this study we use the one that yielded the best performance in the previous study, *column-entropy distance*, defined below. This distance measure is computed from the (column normalized) *diffusion character matrix* of a network.

Diffusion character matrices (or DC matrices) place a parameterized coordinate system on a network that is closely tied to the behaviour of random walks on the network. Diffusion character matrices are introduced and several of their mathematical properties are derived in [6]. The diffusion character matrix of a network is the Neumann series of a scalar multiple ($\frac{1}{2}$ for this study) of the column-normalized adjacency matrix of a network, it is known as *Random Walk with Restart* in clustering (see [14] and [15]) and as the standard vector space basis for *Personalized PageRank* for search engines (see [16] and [17]). The quickest way to explain the modified DC matrices of this study, and their utility is with the following analogy. A different gas is released at each node of the network, adding one unit of gas at each node in each time step. All gas of a single type at a node is divided evenly among its neighbours and itself in each time step and then half of each type of gas at each node is absorbed, meaning the total amount of gas of each type decays exponentially. The juxtaposition of arithmetic addition and exponential decay of gas cause the system to rapidly settle to a fixed distribution of each gas at each node. The amounts $a_{i,j}$ of gas j at node i are the entries of the DC matrix.

The *Column-Entropy distance* (CE distance) between networks is computed as follows. First the columns (the amounts of a single gas across all network nodes) of the DC-matrices are normalized by scaling the matrix by a constant. The entropy of each column is then computed, yielding an *entropy vector* for the network. The entropy associated with a column represents the evenness of distribution of other nodes as destinations of random walks beginning at the node indexing the column. (Note: entropy of a unit vector v = $\sum_{i=1}^{n} -v_i log_2(v_i)$.) The entropies are then sorted into decreasing order to create the *sorted entropy vector* for a network. The CE-distance between two networks is the Euclidean distance between their sorted entropy vectors. This sorting step is a fast method of approximating correspondence between nodes in the two networks. A more in depth explanation of DC-matrices and the CE-distance used in this study can be found in [5].

1) Nonlinear Projection: Nonlinear projection [18] is a visualization technique used to visualize high dimensional data. It is an evolutionary form of *non-dimensional multimetric scaling* [19] or *multidimensional scaling* [20]. The goal is to provide a projection of points from a high-dimensional space into a two-dimensional space that distorts the inter-point distances as little as possible. The projection forms a visualization of the higher dimensional data set. The original form of NLP used evolutionary computation to minimize the squared error between the distance matrix of the original data and the Euclidean distance matrix of the projection of the data into two dimensions. The representation consists of a simple list of point coordinates. The optimization problem of locating a good projection is treated as a standard real-valued evolutionary optimization.

In this study we maximize the *Pearson correlation* (Equation 2) between the entries of the distance matrices for the original and projected data sets.



Profile 1

Profile 2

Profile 3

Profile 4

Profile 5

Profile 6

Fig. 4. The plots above are the best results, over thirty replicates, of the evolutionary algorithm attempting to match each of the six epidemic profiles. The epidemic profile is shown in black while 95% confidence intervals for the mean of 50 simulated epidemics are shown in blue. The tracks of the individual epidemics are shown in gray in the background.

$$cor = \frac{\sum_{i=1}^{n}(x_i - \overline{x})(y_i - \overline{y})}{(n-1)s_x s_y} \quad (2)$$

Where for $z \in \{x, y\}$, $\overline{z}$ denotes the sample mean and s_z denotes the sample variance.

An issue with minimizing squared error between original and projected distances is that the optimizer must both get the inter-point distances right and, in order to do so, must estimate the scale of the data. This is possible but is an added, unnecessary complication. Maximizing the Pearson correlation, given in Equation 2, of the distance matrix of the high dimensional points with the analogous distance matrix of the projected points in the plane avoids this problem. Pearson correlation is invariant under translation and scaling of either of the sets of points and so permits the evolutionary algorithm to solve the problem of relative distance without worrying about scale, though it yields a scale-free result that shows only relative position of the projected points.

The evolutionary algorithm used to perform nonlinear projections uses a population of ten tentative projections stored as lists of points (x, y). The points are initially generated to lie within the unit square with corners $(0, 0)$ and $(1, 1)$. The model of evolution is tournament selection of size seven. Variation operators are two-point crossover of the lists of points (points are treated as atomic objects) and two mutation operators, each used 50% of the time. The first mutation operator randomly replaces a point with a new point selected uniformly at random within the unit square. The second adds a Gaussian random variable with a standard deviation of 0.1 to both coordinates of a point. From 1 to 3 mutations are performed on any new structure, with the number selected uniformly at random. The algorithm is run for 100,000 mating events.

IV. Results and Discussion

Figure 4 shows that the algorithm was able to find networks that provided adequate matches to all six network profiles. Profiles two and six were the easiest to match; they obtained the best fitness values and a large fraction of the runs generated results similar to those shown. In the experiments using profile four a majority of the runs ended with networks whose average behavior was a good fit to the profile. Profile one had many good results, but they were not a majority of the runs. Profiles three and five were the most challenging; the results shown in Figure 4 for these profiles are exceptionally good fits in relation to the others. Juxtaposing this qualitative analysis with the exemplary fitness traces in Figure 2 we see that the rate of good outcomes agrees with both the final fitness and the degree of disruption after recentering. Profile two has a relatively low rate of such disruption and profile six has a very low rate. Note that the example fitness trace for profile three is one in which recentering blundered into a worse part of the space at roughly mating event 260,000 (remember the units on the x-axis of Figure 2 is 200's of mating events so the 260,000th mating event is x-axis tick 1300). The best network in this run was the one used in the sixth-to-last recentering.

Table I shows 95% confidence intervals for the mean CE distance between graphs in the same and different categories. Note that the matrix is symmetric about the main diagonal and the duplicative entries below the diagonal are included to permit comparison of entire rows and columns. A non-linear projection of the full CE-distance matrix for all 180 best-of-run evolved networks, giving an approximate visual representation of the scatter of the networks in network space, is given in Figure 5. The most remarkable result in this table and figure is that profile six (the one that was easiest to fit) has no measurable scatter within the set of 30 networks produced by that experiment. The substantial scatter for profile 6 networks in the non-linear projection results entirely from their juxtaposition with networks evolved to match other profiles. The other five profiles have confidence intervals that overlap slightly, in agreement with the non-linear projection, but there is no profile whose networks are distributed evenly through the space. Given that several to most of the networks do not fit their profiles well, the scatter shown in Figure 5 suggests that the algorithm is producing substantially different categories of graphs for the different profiles.

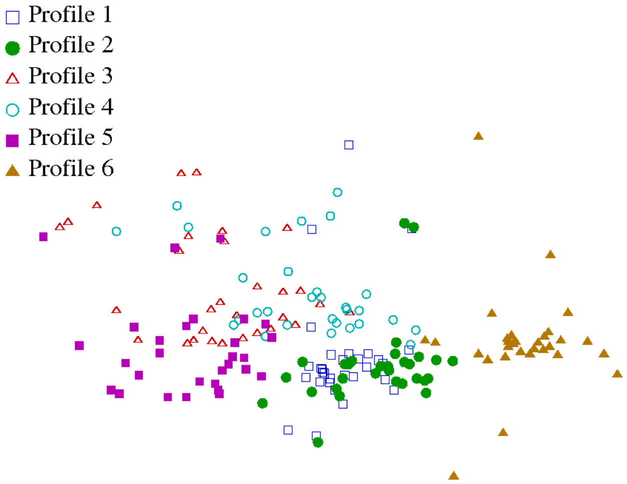


Fig. 5. Nonlinear projection of the distance matrix of Column-Entropy distances between individual best-of-run graphs for all six epidemic profiles. The correlation coefficient for this projection is $\rho = 0.978$ suggesting high fidelity of representation.

V. Conclusions and Next Steps

Figure 4 demonstrates that the recentering-restarting evolutionary algorithm is able to fit networks to several distinct epidemic profiles. These results expand the scope of this line of investigation to include searching the space of networks for a broader variety of profiles than those in [4]. This study also applies diffusion character based analysis to the resulting networks. This is the first time this has been done for networks evolved to match epidemic profiles.

The author's studies on network evolution before this one all use network editing operators (edge-swap, simplexifica-

TABLE I
SHOWN ARE 95% CONFIDENCE INTERVALS ON THE MEAN DISTANCE BETWEEN GRAPHS IN EACH PAIR OF EXPERIMENTS. DIAGONAL ENTRIES ESTIMATE THE DISTANCE BETWEEN GRAPHS IN THE SAME EXPERIMENT.

	Profile 1	Profile 2	Profile 3	Profile 4	Profile 5	Profile 6
Profile 1	2.006 ± 1.115	2.248 ± 1.038	3.856 ± 1.371	2.937 ± 1.055	3.723 ± 1.229	4.584 ± 0.908
Profile 2	2.248 ± 1.038	1.966 ± 0.981	4.292 ± 1.497	3.120 ± 1.257	4.183 ± 1.358	4.130 ± 1.152
Profile 3	3.856 ± 1.371	4.292 ± 1.497	2.765 ± 1.134	2.781 ± 1.223	2.837 ± 1.121	6.254 ± 1.420
Profile 4	2.937 ± 1.055	3.120 ± 1.257	2.781 ± 1.223	2.407 ± 1.106	3.487 ± 1.229	4.864 ± 1.362
Profile 5	3.723 ± 1.229	4.183 ± 1.358	2.837 ± 1.121	3.487 ± 1.229	2.294 ± 1.050	6.964 ± 1.181
Profile 6	4.584 ± 0.908	4.130 ± 1.152	6.254 ± 1.420	4.864 ± 1.362	6.964 ± 1.181	0.000 ± 0.000

tion, etc.) These could easily be combined with the toggle and edge deletion operation to create richer sets of generative operations for network evolution. Determining which sets of operations are appropriate for each type of network search problem is another potentially rich source of future work.

This study joins several others in confirming the utility of the recentering-restarting evolutionary algorithm for searching spaces of networks. Earlier studies using this type of evolutionary algorithm [10], [5], [4] used a variety of different evolvable network representation and different fitness functions, and so a broad demonstration of the utility of recentering-restarting algorithms now exists.

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, or the choice of epidemic model, was performed. 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. Other epidemic models 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, categorizing the nodes as male and female, by age, or by susceptibility potentially permits much higher fidelity of modeling.

While the network-evolution technology presented here is applied to the problem of finding networks likely to yield a particular epidemic profile, the network evolution technology is general. Writing appropriate fitness functions will permit a search for network-structured solutions to a variety of problems, ranging from theoretical graph theory to robust network design.

REFERENCES

[1] J. Wylie and A. Jolly, “Patterns of chlamydia and gonorrhea infection in sexual networks in Manitoba, Canada.” *Sexually Transmitted Diseases*, vol. 28, no. 1, pp. 14–24, 2001.

[2] W. J. Edmuns, C. J. O’Calliaghan, and D. J. Nokes, “Who mixes with whom? A method to determine the contact patterns of adults that may lead to the spread of airborne infections,” *Proceedings of the Royal Society of London(B)*, vol. 264, no. 1384, pp. 949–957, 1997.

[3] S. Aral *et. al.*, “Sexual mixing patterns in the spread of gonococcal and chlamydia infections.” *American Journal of Public Health*, vol. 6, no. 89, pp. 825–833, 1999.

[4] D. Ashlock and E. Shiller, “Fitting contact networks to epidemic behavior with an evolutionary algorithm,” 2011, accepted to the 2011 IEEE Symposium on Computational Intelligence in Bioinformatics and Computational Biology.

[5] D. Ashlock and C. Lee, “Characterization of extremal epidemic networks with diffusion characters,” in *Proccedings of the 2008 IEEE Symposium on Computational Intelligence in Bioinformatics and Computational Biology*. Piscataway NJ: IEEE Press, 2008, pp. 264–271.

[6] ——, “Diffusion characters: Breaking the spectral barrier,” in *Proceedings of the 21st Canadian Conference on Electrical and Computer Engineering*. Piscataway NJ: IEEE Press, 2008, pp. 847–850.

[7] F. Hoppensteadt and C. Peskin, *Mathematics in Medicine and the Life Sciences*. New York, NY: Springer-Verlag, 1992.

[8] D. Ashlock and F. Jafargholi, “Evolving extremal epidemic networks,” in *Proceedings of the 2007 IEEE Symposium on Computational Intelligence in Bioinformatics and Computational Biology*. Piscataway NJ: IEEE Press, 2007, pp. 338–345.

[9] ——, “Representations for the evolution of extremal epidemic networks,” *International Journal of Information Technology and Intelligent Computing*, vol. 2, no. 3, pp. 11–20, 2007.

[10] ——, “Representations for the evolution of extremal epidemic networks,” in *Proceedings of the 2008 World Congress on Computational Inteligence*. Piscataway NJ: IEEE Press, 2008, pp. 660–667.

[11] V. Latora, A. Nyamba, J. Simpore, B. Sylvette, and S. Diane, “Network of sexual contacts and sexually transmitted hiv infection in Burkina Faso,” *Journal of Medical Virology*, vol. 78, pp. 724–729, 2006.

[12] F. Liljeros and C. R. E. et. al., “The web of human sexual contacts.” *Nature*, vol. 411, pp. 907–908, 2001.

[13] G. Syswerda, “A study of reproduction in generational and steady state genetic algorithms.” in *Foundations of Genetic Algorithms*. Morgan Kaufmann, 1991, pp. 94–101.

[14] D. C. J. Sun, H. Qu and C. Faloutsos, “Neighborhood formation and anomaly detection in bipartite graphs.” in *Proceedings of Proceedings of Fifth International Conference on Data Mining*, ser. ICDM’05, 2005, pp. 418–425.

[15] H. Tong, C. Faloutsos, and P. Jia-Yu, “Fast random walk with restart and its applications,” in *Proceedings of the Sixth International Conference on Data Mining*, ser. ICDM ’06, 2006, pp. 613–622.

[16] S. Brin and L. Page, “The anatomy of a large-scale hypertextual web search engine,” in *Proceedings of the Seventh International Conference on World Wide Web 7*, ser. WWW7, 1998, pp. 107–117.

[17] G. Jeh and J. Widom, “Scaling personalized web search,” in *Proceedings of the 12th international conference on World Wide Web*, ser. WWW ’03, 2003, pp. 271–279.

[18] J. Schonfeld and D. Ashlock, “Filtration and depth annotation improve non-linear projection for rna motif discovery,” in *Proceedings of the 2006 IEEE Symposium on Computational Intelligence in Bioinformatics and Computational Biology*. Piscataway NJ: IEEE Press, 2006, pp. 352–359.

[19] P. Legendre and L. Legendre, *Numerical Ecology, 2nd English Edition*. New York: Elsevier, 1998.

[20] I. Borg and P. Groenen, *Modern multidimensional scaling. Theory and applications.* New York: Springer, 1997.