Evolving the Curve

Michael Dubé
Dept. of Computer Science
Brock University
St. Catharines, ON, Canada
md12ol@brocku.ca

Sheridan Houghten
Dept. of Computer Science
Brock University
St. Catharines, ON, Canada
shoughten@brocku.ca

Daniel Ashlock
Dept. of Mathematics and Statistics
University of Guelph
Guelph, ON, Canada
dashlock@uoguelph.ca

James Alexander Hughes
Dept. of Computer Science
St. Francis Xavier University
Antigonish, NS, Canada
jhughes@stfx.ca

***Abstract*—Evolutionary algorithms are used to generate personal contact networks, modelling human populations, that are most likely to match a given epidemic profile. The Susceptible-Infected-Removed (SIR) model is used and also expanded upon to allow for an extended period of infection, termed the SIIR model. The networks generated for each of these models are thoroughly evaluated for their ability to match nine different epidemic profiles. The addition of the SIIR model showed that the model of infection has an impact on the networks generated. For the SIR and SIIR models, these differences were relatively minor in most cases.**

I. Introduction

This paper focuses on the ability to generate personal contact networks, representing physical connections between community members, which satisfy the data about the number of infections per time period. A personal contact network is the foundation of an epidemic model in which an epidemic spreads along the links of the network. This study compares two models of disease spread, in preparation for incorporating an asymptomatic state to match the behavior of SARS-Covid-2. The approach of employing a generative solution to a test problem is known as *graph induction* which has a variety of applications [8], [10], [13]. The representation used within this paper is known as the Local THADS-N generative representation; the metric used to evaluate the performance of a network is epidemic *profile matching*, introduced in [5]. The representation is described in detail in Section III.

A. Organization of Paper

The remainder of this paper is organized as follows. Section II provides background information on graphs and epidemics. Section III describes the representation. Section IV gives the experimental design and discusses parameter settings. Section V presents and discusses results. Section VI presents conclusions and identifies possible directions for future research.

II. Background

A. Graph Theory

The personal contact network used in this work is implemented as a combinatorial graph. Individuals are the *vertices* of the graph and the connections between individuals *edges*. The terms network and graph are used interchangeably within this paper. A graph G is defined be specifying a set of vertices V and edges E, denoted $G(V, E)$. An edge is represented as an unordered pair of vertices $\{p, q\}$. Only undirected graphs are used: infection can pass in either direction. A path from vertex p to vertex q on graph G is a sequence of edges from E which connect p and q. The *distance* from p to q is the length of the shortest path which connects p and q.

B. The Models of Infection Used

The Susceptible-Infected-Removed (SIR) model of infection [11] provides a simple model for the simulation of epidemics. In this model the population is divided into three mutually exclusive groups: those still able to be infected by the epidemic are *susceptible*, those that currently have the epidemic are *infected*, and those that were previously infected are *removed* (due to immunity or death). An epidemic begins by choosing one individual within the population to be infected. The epidemic then spreads probabilistically along edges of the network. An individual has a probability α of being infected by each adjacent infected population member, with the probabilities evaluated independently. In the SIR model, the epidemic disease lasts a single time step within an infected individual. The current study also allows for the infected stage to last two time steps, which we term *SIIR*; conceptually, this relates to a situation in which an individual is contagious for a longer length of time, thereby providing them an additional timestep in which they can infect others. This paper compares graphs evolved to match epidemic profiles with the SIR and SIIR models to assess the degree of influence the model has on the graph that arises. Another reason the SIIR model was chosen was in preparation for incorporating the *SEIR* model in which *Exposed* individuals have contracted the disease yet are not infectious; this is akin to the incubation period of a virus.

C. Profile Matching (PM) Problem

Introduced in [5], epidemic profile matching begins with defined epidemic behavior on a human population to determine if networks likely to permit similar behavior can be generated. An epidemic profile is specified by the number of individuals infected at each time step of an epidemic simulation. There is no evidence to suggest that a particular network is ideal for any given epidemic profile, therefore the goal of the epidemic profile matching (PM) problem is to find networks likely to generate behavior resembling the profile. The nine profiles used to test this problem, from [5], are shown in Fig. 1. These profiles were chosen to allow for comparison to previous work as well as to provide a range of potential epidemic behaviour.

978-1-7281-9468-4/20/$31.00 ©2020 IEEE

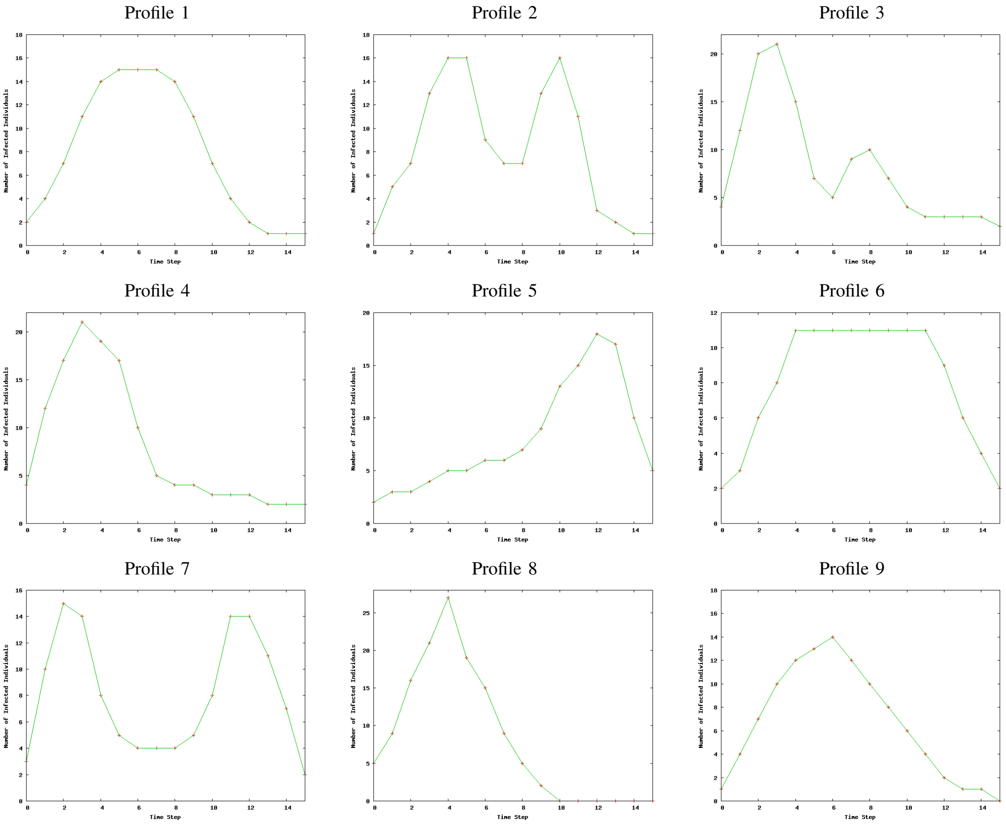


Fig. 1. Epidemic profiles representing time step vs. number of infected individuals during that time step.

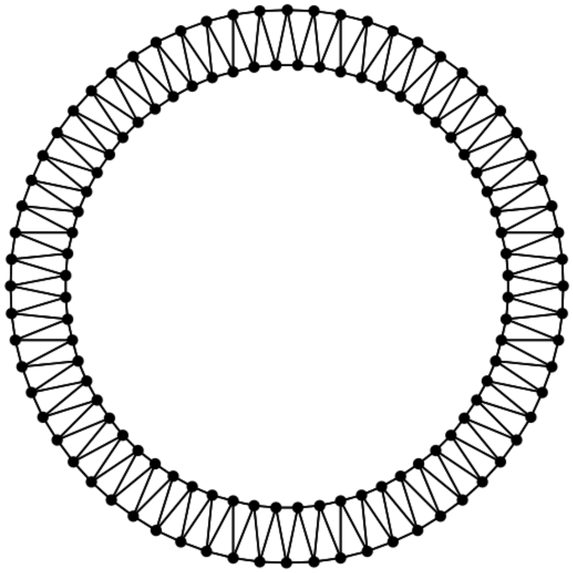


Fig. 2. Initial graph with 128 vertices on which to apply the string of edge operations. Each vertex has two edges to the two preceding nodes as well as two edges to the two proceeding nodes in the ring.

III. The Local THADS-N Representation

The Local THADS-N representation is a generative representation that creates networks from a starting network through a series of editing commands that modify the connections in the network. These are called *edge-editing operations*. A generative solution is chosen because it permits the inclusion of domain information, e.g. a reasonable number of edges, in the initial graph. Use of a generative representation also permits search of the space of graphs with a simple linear structure, the list of editing commands. Examples of other generative representations are given in [8], [10], [13]. Evidence of the effectiveness of generative solutions appears in [12]. The initial graph used in this study, to which edits are applied, is shown in Fig. 2. This graph was chosen because earlier research [3], [5] has demonstrated that graphs having vertices with degree 4-5 are desirable for the test problems. Other work on network evolution appears in [1]–[4], [6].

A. Edge Operations

Given a graph $G(V, E)$ and the vertices p, q, r, and s from the set V the existing operations are defined below. Strings of these operations make up the chromosomes used in the evolutionary algorithm defined in Section IV-A

- **Toggle**(p**,** q)**:** If edge $\{p, q\}$ is in E then remove $\{p, q\}$ from E, otherwise add $\{p, q\}$ to E.
- **Local Toggle**(p**,** q**,** r)**:** If edge $\{p, q\}$ and $\{q, r\}$ are in E then **Toggle**(p, r).

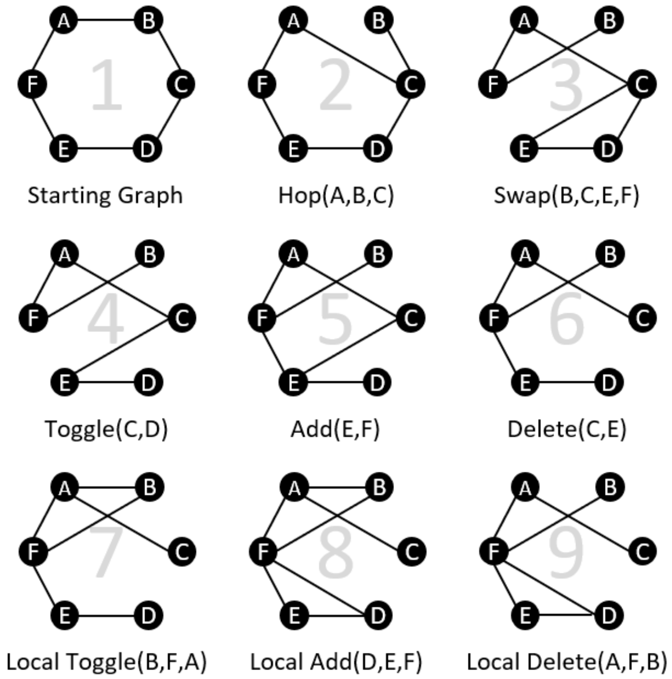


Fig. 3. Examples of operators included in the Local THADS-N representation. The figure shows eight of the nine operations being applied sequentially to an initial six cycle graph. The operations are applied in reading order such that the final graph is the result of applying all eight operations. The null operation is omitted as it does not change the graph.

- **Hop(**p**,** q**,** r**):** If edge $\{p, q\}$ and $\{q, r\}$ are in E and edge $\{p, r\}$ is not in E then remove edge $\{p, q\}$ from E and add edge $\{p, r\}$ to E.
- **Add(**p**,** q**):** If $\{p, q\}$ is not in E then add $\{p, q\}$ to E, otherwise do nothing.
- **Local Add(**p**,** q**,** r**):** If edge $\{p, q\}$ and $\{q, r\}$ are in E then **Add(**p, r).
- **Delete(**p**,** q**):** If $\{p, q\}$ is in E then remove $\{p, q\}$ from E, otherwise do nothing.
- **Local Delete(**p**,** q**,** r**):** If edge $\{p, q\}$ and $\{q, r\}$ are in E then **Delete(**p, r).
- **Swap(**p**,** q**,** r**,** s**):** If $\{p, q\}$ and $\{r, s\}$ are the only edges between p, q, r and s then remove $\{p, q\}$ and $\{r, s\}$ from E and add $\{p, s\}$ and $\{q, r\}$ to E.
- **Null():** Do nothing.

IV. Experimental Design

A. Evolutionary Computation

A steady state evolutionary algorithm [15] is used to generate the solutions, which are strings of edge operations. All variables with respect to system design were determined empirically.

A population of 1000 chromosomes is used, each of which contains a string of 256 Local THADS-N edge operations. A given string of operations applied to the initial 128-vertex graph in Fig. 2 produces a candidate solution to the test problem. The chromosomes are initially generated at random based upon the probabilities provided to each of the operations via the program parameters. The chromosomes then undergo 40,000 mating events with output every 400 events. Each mating event consists of a round of tournament selection, crossover and mutation. Tournament selection selects 7 chromosomes at random from the population, evaluates their fitness and replaces the two chromosomes with the worst fitness by copies of the two chromosomes with the best fitness. These two copies then undergo two-point crossover, and mutation occurs on 1-3 of the operations within that chromosome, replacing them with new commands chosen by the same probability distribution. The choice of 1-3 mutations is randomly determined with each choice being equiprobable. Finally, fitness is recalculated for the children. After evolution the candidate solution with the best fitness from the whole population is saved. The process is repeated 30 times for each parameter setting (PS), and the entire procedure is repeated on each of the nine epidemic profiles in Fig. 1.

In order to determine which solutions should be favoured for evolution the profile matching fitness is calculated by simulating epidemics on the personal contact networks from [16]. In each epidemic the vertex with the lowest index, *patient zero*, is marked infected and the epidemic is permitted to spread along edges from vertex to vertex. These epidemics have probability $\alpha = 50\%$ of spreading to susceptible individuals via edges in the graph; each of these probabilities is calculated independently. It is important to note that this fitness measure does not indicate the absolute quality of a network. Instead, it measures the relative quality of a network, permitting successive candidate solutions to converge to networks which are more likely to create epidemics satisfying the problem.

This fitness function determines a solution's fitness by simulating 50 epidemics. It compares the number of infected individuals at each time step of each epidemic with the expected number of infected individuals in the profile to be matched, to calculate the sum squared error (SSE) of a solution. The 50 SSE measurements are then sorted in increasing order $E_1 \leq E_2 \leq \ldots \leq E_n$. This ordering is used to determine the fitness of a graph G in the form of a linearly weighted sum of the measurements according to *fit(G)* $= \sum_{i=1}^{n} \frac{E_i}{i}$.

As the SSEs are sorted, this allows for the fitness function to be most impacted by those simulated epidemics which most accurately resemble the known epidemic profile being considered. In order to provide a fitness value for a network which can be compared to other networks the fitness is calculated, without weighting, after execution.

B. An Entropic Pseudometric for Comparing Graphs

One of the major hypotheses under test in this study is that, within the same epidemic profile, changing the model of disease spread will cause the graph induction system to produce substantially different graphs. To document this, we need a way to compare graphs that is not obfuscated by the many irrelevant variations in structure. A pseudometric that has these qualities is the Column-Entropy distance(CE-

TABLE I
THE SETS OF LOCAL THADS-N EDGE OPERATION PROBABILITIES FROM [9] FOR CONSTRUCTING SOLUTIONS USING THE EVOLUTIONARY ALGORITHM DESCRIBED IN SECTION IV-A

Experiment	Toggle	Hop	Add	Delete	Swap	L-Toggle	L-Add	L-Delete	Null
PS1	0.2528	0.0142	0.0087	0.2138	0.0021	0.2267	0.2228	0.0079	0.0509
PS2	0.0056	0.0016	0.0133	0.0032	0.0272	0.0177	0.8115	0.0214	0.0985
PS3	0.2713	0.0041	0.0133	0.0129	0.0311	0.0446	0.5675	0.0068	0.0484
PS4	0.0044	0.0419	0.0082	0.0149	0.0135	0.3141	0.5054	0.0366	0.0611
PS5	0.0090	0.0002	0.3233	0.0183	0.0020	0.0128	0.4968	0.0249	0.1128
PS6	0.4925	0.0061	0.0120	0.0517	0.0083	0.0052	0.2846	0.0127	0.1268
PS7	0.0197	0.0795	0.7238	0.0001	0.0481	0.0175	0.0393	0.0038	0.0684
PS8	0.0084	0.0159	0.0323	0.0172	0.0021	0.0046	0.4702	0.3775	0.0718

distance) [14]. Computation is based on the simulated diffusion of a collection of different gasses, one per vertex, with absorption of all gasses present taking place at a low rate at each vertex. This process converges rapidly as gas is added arithmetically but decays exponentially via absorption. The result is a matrix, with rows indexed by vertices and columns indexed by gasses, giving the amount of each gas at each vertex. Columns are normalized to sum to one and then the entropy of each column is 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. Entropies are then sorted into decreasing order to create sorted entropy vector for a network. The CE-distance between two networks is the Euclidean distance between their sorted entropy vectors. The sorting step is a fast method of approximating correspondence between nodes in the two networks. A more detailed explanation of this, and other pseudometrics on networks, appears in [14]. A pseudometric is a distance measure with the property that two dissimilar objects can be at distance zero from one another – something that did not occur in practice in this study, meaning that the CE-distance is functionally a metric in this study.

V. RESULTS AND DISCUSSION

Taking the network with best (lowest) fitness from each run of the evolutionary algorithm results in 30 graphs for each (parameter setting, profile) pair. These were used to generate box and whisker plots of the profile matching fitness on the nine profiles using the SIIR model of infection, shown in Figure 4. Additionally, results from previous work using the SIR model of infection are included [9] to investigate the impact of increasing the infectious period of an epidemic. It is clear that the fitness values achieved remain consistent between the SIR and SIIR epidemic models. The confidence intervals achieved by both models overlap, with the parameter settings having similar impacts on performance regardless of the model chosen. This demonstrates that the addition of the SIIR model does not have a tangible impact on the overall fitness of the networks generated. However, although the fitness values are similar other differences can exist within the networks, which will be investigated in the following sections.

TABLE II
THE PARAMETER SETTING WITH THE LOWEST (BEST) MEAN FITNESS ACROSS 30 RUNS FOR EACH PROFILE AND MODEL OF INFECTION USED.

Profile	SIR Best	SIR Mean	SIIR Best	SIIR Mean
1	PS3	7.7314	PS2	7.7023
2	PS3	9.8480	PS3	9.8514
3	PS3	8.7472	PS3	8.7454
4	PS3	7.7277	PS3	7.7803
5	PS2	9.6765	PS2	9.5650
6	PS2	8.5336	PS2	8.4512
7	PS2	9.9484	PS2	9.8524
8	PS7	7.1829	PS7	7.1637
9	PS2	7.3203	PS2	7.3143

A. Graph Visualizations

To investigate the impact of the SIIR model on the networks generated, visualizations of the graphs were created. The network with best (lowest) fitness from the parameter setting with the lowest mean fitness is chosen from each of the epidemic models studied. See Table II for the parameter settings and mean fitness corresponding to the visualizations being compared. To aid in analyzing the differences between networks the nodes were coloured as follows: patient zero is red, nodes 1-31 are cyan, nodes 32-63 are orange, nodes 64-95 are yellow, and nodes 96-127 are green. Visuals for profiles 2, 4, and 6 are provided respectively in Figures 5, 6, and 7.

Looking at these visualizations there seem to be clear differences between the networks on the two models of infection. For profile 2 the SIR model results in significant intermixing of the four node colours throughout the network. The SIR model also retains a lot of the chain in the initial network from Figure 2, most notably with the green nodes at the top left and cyan nodes at the bottom right. In contrast the SIIR model results in much less intermingling of the four colours with four distinct clusters remaining intact in the evolved network while much of the chain structure is lost.

The retained chain structure is a feature of both networks generated on profile 4 in Figure 6, although the location of the chain within the resultant network is different between models. The SIR model contains distinct chains in the blue and orange nodes, while the SIIR model features chains in the yellow and orange nodes. Flipping one of the networks reveals that the overall structure of the network is similar, with the non-chain section of the network resembling a large well connected cluster of all node colours and patient zero.

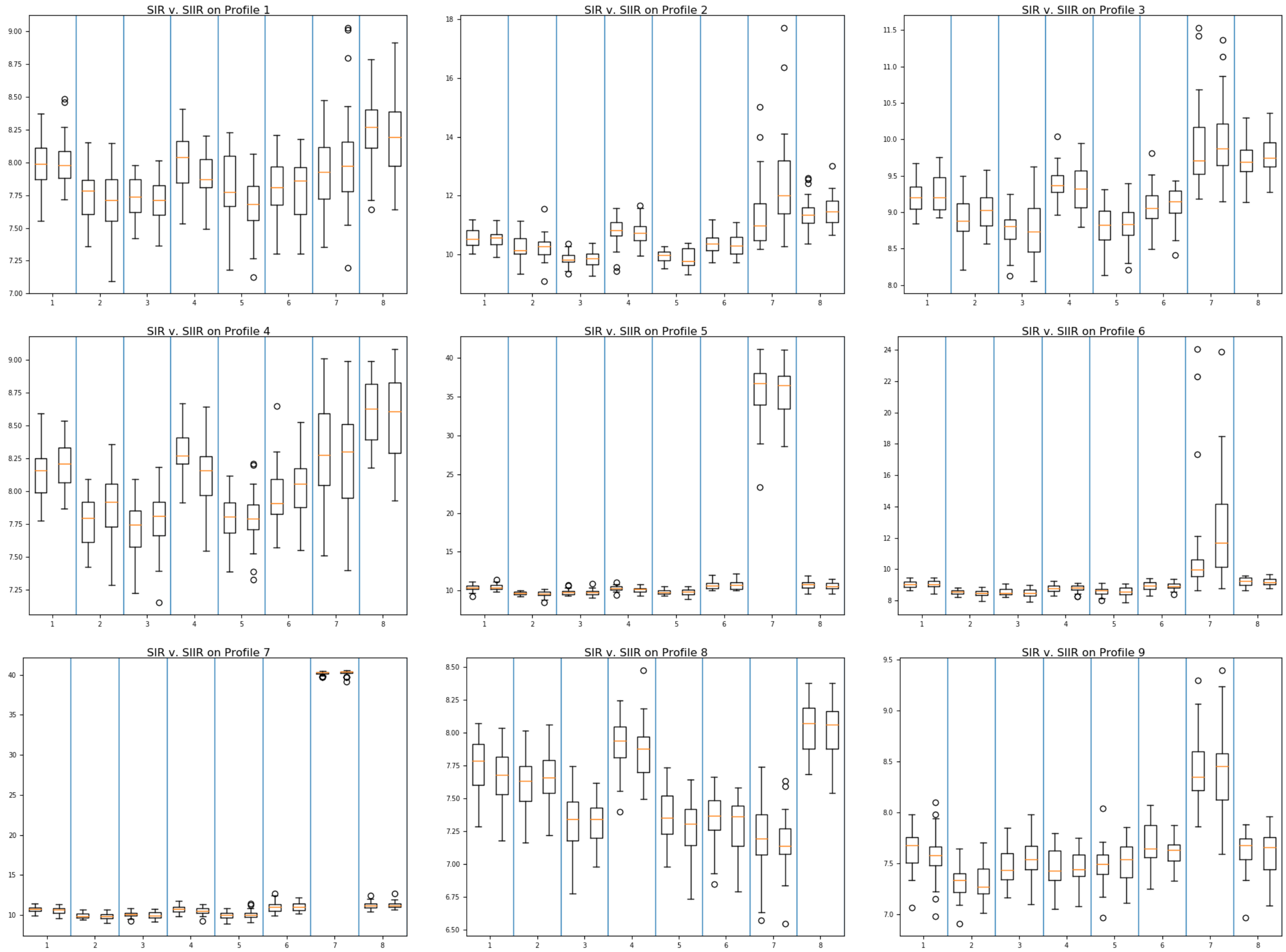


Fig. 4. Box and whisker plots of the profile matching fitness achieved on 30 runs of the evolutionary algorithm using the SIR and SIIR epidemic models. The eight columns correspond to the eight parameter settings with the left box plotting the fitness using the SIR model, and the right the SIIR model within each column.

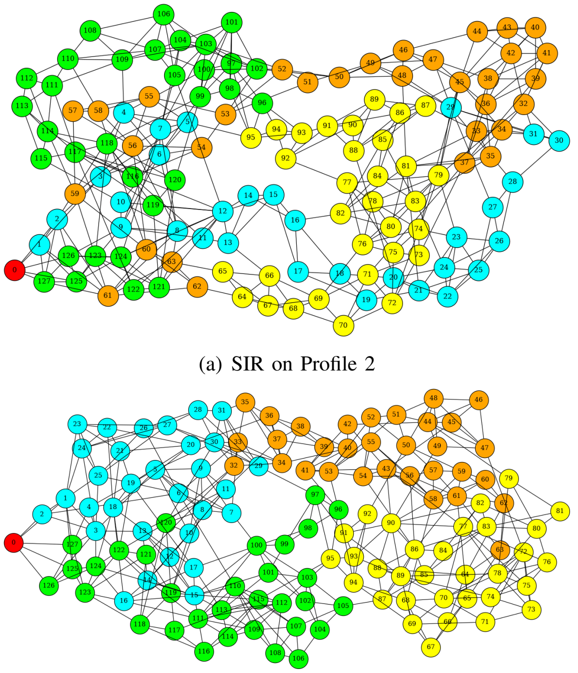
The networks generated using profile 6 in Figure 7 feature large sections of the initial structure of the network. The SIR model is made almost entirely of the chain structures with most of the orange, yellow and green nodes being part of the chain. The SIIR network also has large sections of the chain retained for green, blue, and orange nodes, although less notably than in the SIR model. Also, the SIIR model allows for more mingling between nodes of different colours than the SIR model.

The differences between networks generated using different models of infection are similar for the remaining profiles, although the changes are not as stark as for the networks from the above profiles. A more thorough investigation is necessary to gain insight into the patterns that exist on the plethora of networks generated in this study.

B. An Entropic Pseudometric for Comparing Graphs

The final structure of the networks generated can fluctuate based on the various parameter settings, profiles and epidemic models used within the study. Therefore the column entropy distance is computed in a pair-wise manner to determine the distance between any pair of networks. The network used to compute this value for any given system configuration is the network which achieved the best fitness across the runs. These values were used to generate heat maps of the column entropy distance between and within the two epidemic models; see Figure 8. Dark blue represents a difference of zero while bright yellow indicates a high distance.

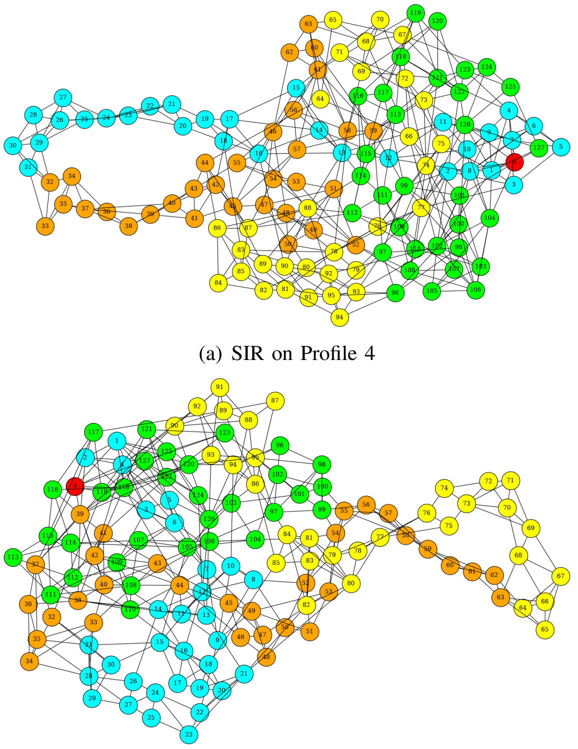
The first heat map compares the 72 networks generated using the SIR model with the 72 generated using the SIIR model. Different patterns emerge across system configurations. The top-left quadrant demonstrates that the networks are somewhat impacted by the epidemic model being used, with some cells being darker blue and others approaching the median value. In contrast the center, specifically comparing profile 5 to 7 across models, features networks with minimal column entropy distance. Furthermore, profile 8 and 9 are the brightest rows/columns revealing the large variability between the networks realized under the two models. Most notably, profile 9 under SIR results in networks furthest from those generated using SIIR on profile 8, with this section of the



(a) SIR on Profile 2

(b) SIIR on Profile 2

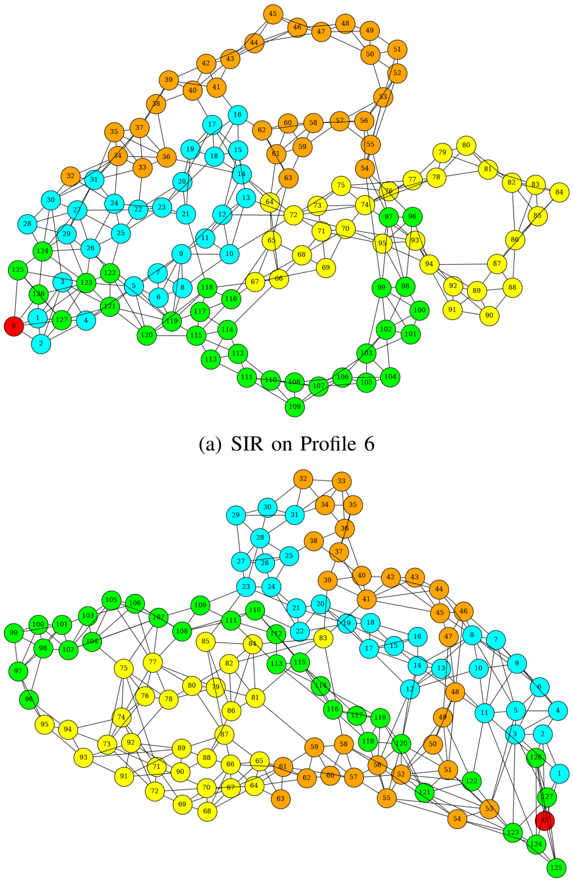
Fig. 5. The visualization of the graph with lowest fitness on profile 2 generated under the specified model of infection. See Table II for parameter setting and fitness values.



(a) SIR on Profile 4

(b) SIIR on Profile 4

Fig. 6. The visualization of the graph with lowest fitness on profile 4 generated under the specified model of infection. See Table II for parameter setting and fitness values.



(a) SIR on Profile 6

(b) SIIR on Profile 6

Fig. 7. The visualization of the graph with lowest fitness on profile 6 generated under the specified model of infection. See Table II for parameter setting and fitness values.

map being brightest.

The second heat map compares networks generated using the SIR model with themselves, and the third map does this for the SIIR model. The patterns from the first heat map are largely apparent in these maps as well with only slight variations. This provides evidence that the networks generated are dominated by the profile and to a lesser degree the parameter setting being used to select edge operations. Therefore, the variable having the least impact is in fact the model of infection chosen, although small variations exist between the maps. Under the SIR model the top-left quadrant demonstrates less variability between networks from settings on profiles 1-4. The center of the map is noticeably brighter than the first figure, meaning that the distance between networks generated within the SIR environment actually differ *more* than those between the models, an unexpected result.

The final heat map is the brightest of the three. This means that the networks created using the SIIR epidemic actually differ more from each other than from those generated using the SIR model. Once again the most significant variations exist for the networks created using profile 8 and 9. These profiles are comprised of epidemic curves that are heavily weighted towards the start of an outbreak. This would favour sporadic epidemic behaviour in which the outcome of the epidemic depends heavily on where and how quickly the virus spreads in the first few time steps. This likely contributes to the variability between networks observed in the heat map. The rows and columns that are coloured almost entirely yellow and green in profile 5 and 7 under PS7 are due to the reliance on the add operation within that parameter setting. This causes the epidemics to quickly spread and infect the entirety of a population early on. However, these profiles feature a significant portion of their infections later in the outbreak so the fitness is impacted significantly, leading to the large column entropy distance.

C. SIR on SIIR, etc. Epidemic Profiles

The final method of comparison between the epidemic models involves the epidemic profiles or epidemic curve that result from simulating an epidemic. The network with the lowest mean fitness, across all parameter settings, is used for each epidemic model. This provides two networks per epidemic profile. On each of the networks 500 SIR and 500 SIIR epidemics are simulated; the SIR and SIIR epidemic with lowest fitness for each model is then plotted against the profile being evolved to. This plot is available for profile 1 in Figure 9. The networks generated using the SIR model result in curves which most closely resemble the profile being evolved to, irrespective of the type of epidemic being actualized. Although, the network that came from the SIIR model results in two epidemics which overshoot the epidemic curve but in a similar manner. The SIIR epidemic provides a steeper and taller curve than the SIR epidemic's more jagged and delayed peak. This is likely because the increased infection length allows for greater and faster spread of the epidemic.

VI. Conclusions and Future Work

The addition of the SIIR model of infection was hypothesized to result in differences in the networks generated using the system described above. This addition did provide evidence that the model of infection has an impact on the networks generated though the differences are minor in the majority of cases. The algorithm was able to adapt to the SIIR model by matching the fitness achieved by the SIR model in all cases. The visualizations of the networks generated demonstrate that differences in structure are present based on the model used. This structure allows for two different epidemic models to generate the same epidemic curve on their respective network, as shown in Figure 9. Lastly, the column entropy distance heat maps revealed that the profile, parameter setting, as well as the epidemic model all contribute to the networks generated and fitness achieved, although the profile and parameter setting cause the majority of the fluctuation when compared to the model of infection being deployed.

A key limitation of this system is the size of the personal contact networks that can be generated before evolution becomes too costly to be practical. Networks with 128 nodes can model small communities while simulating an epidemic

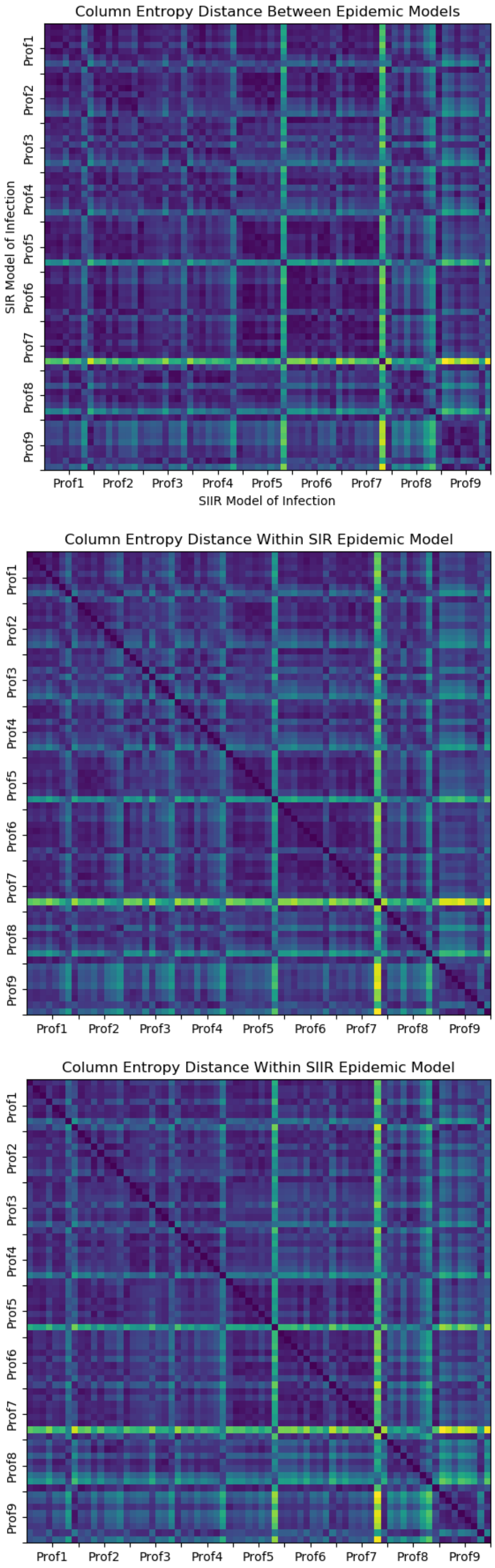


Fig. 8. Heatmaps of column entropy distance between graphs generated using the specified epidemic profile and model of infection. Each profile consists of eight vectors representing results from each of the parameter settings on that profile.

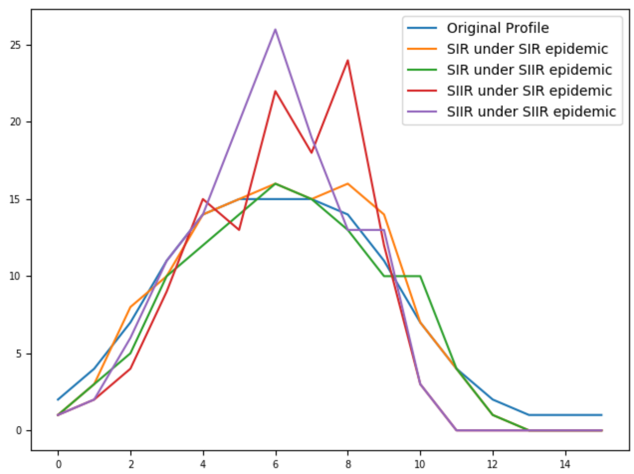


Fig. 9. Number of infected individuals per time step for simulated epidemics on networks generated using the evolutionary algorithm from Section III-A using profile 1. The network with lowest mean fitness evolved using each epidemic model of infection was used to simulate 500 SIR and 500 SIIR epidemics. The epidemic with best fitness is shown.

between communities as part of a personal contact network resembling a city, province, or country. This would also allow for smoother epidemic curves as the network would more closely resemble real world networks with physical and social distance between members of the population, a situation that is not possible with 128 nodes. Moreover, the epidemic curves released by public health professionals often feature rolling averages to handle sporadic epidemic behaviour that changes day-by-day. Larger networks, permitting longer epidemics, would allow for smoother epidemic curves than those realized within this study.

The inclusion of the SIIR model is the first of many possibilities when it comes to exploring more complex epidemic models. A further increase to the length of infectability, such as in [7], along with the inclusion of a presymptomatic or asymptomatic state to the model are also possible. More research should be conducted into potential patterns between the ability of a parameter setting to generate successful networks within various epidemic environments. This relates closely to the continued goal of reducing the complexity of the evolutionary algorithm utilized within this paper to allow for better scalability.

Other modifications could include expansions to the epidemic models used here. For example, the graphs could use directed edges, or the probability of infection could be modelled by using weighted edges. Future work would also benefit from exploring different values for α or replacing a single value with a probability distribution to better resemble real-world virus infectability. Additionally, exploring different initial graphs and their impact on performance could provide new insights and increase the robustness of the software. The personal contact networks from particular countries or communities will undoubtedly have some variation and the software will need to handle this variation. Furthermore, the representation can be applied to new graph-evolution problems which may solve problems in new and interesting domains. Lastly, this model could be tested against real-world data sets from SARS-Covid-2, historical epidemics or outbreaks on campuses, in hospitals or between communities for which data has been recorded and is available. Endless possibilities exist for the expansion of this generative representation.

VII. Acknowledgements

This research was supported in part by ResearchNS' Nova Scotia COVID-19 Health Coalition consisting of Dalhousie University, Dalhousie University Medical Research Foundation, QEII Health Science Centre Foundation, IWK Foundation, Dartmouth General Hospital Foundation, Nova Scotia Health Authority, and Research Nova Scotia.

This research was also supported in part by the Natural Sciences and Engineering Research Council of Canada (NSERC).

References

[1] D. Ashlock and L. Barlow. A class of representations for evolving graphs. In *Proceedings of the 2015 Congress on Evolutionary Computation*, pages 1295–1302, 2015.

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

[3] D. Ashlock and C. Lee. Characterization of extremal epidemic networks with diffusion characters. In *Proceedings of the 2008 IEEE Symposium on Computational Intelligence in Bioinformatics and Computational Biology*, pages 264–271, 2008.

[4] D. Ashlock, J. Schonfeld, L. Barlow, and C. Lee. Test problems and representations for graph evolution. In *Proceedings of the IEEE Symposium on the Foundations of Computational Intelligence*, pages 38–45, 2014.

[5] D. Ashlock and E. Shiller. Fitting contact networks to epidemic behavior with an evolutionary algorithm. In *Proceedings of the 2011 IEEE Symposium on Computational Intelligence in Bioinformatics and Computational Biology*, pages 1–8, Piscataway NJ, 2011. IEEE Press.

[6] D. Ashlock and M. Timmins. Adding local edge mobility to graph evolution. In *Proceedings of the 2016 IEEE World Congress on Computational Intelligence*, Piscataway NJ, 2016. IEEE Press.

[7] Daniel Ashlock, Joseph Brown, and Clinton Innes. *Dawn of the Dead : An Evolvable Linear Representation for Simulating Government Policy in Zombie Outbreaks*. 10 2014.

[8] A. Devert, T. Weise, and K. Tang. A study on scalable representations for evolutionary optimization of ground structures. *Evolutionary computation*, 20(3):453–472, 2012.

[9] Michael Dubé, Sheridan K. Houghten, and Daniel Ashlock. Representation for evolution of epidemic models. In *IEEE Congress on Evolutionary Computation, CEC 2019, Wellington, New Zealand, June 10-13, 2019*, pages 2370–2377. IEEE, 2019.

[10] P. F. Hingston, L. C. Barone, and Z. Michalewicz. *Design by Evolution: Advances in Evolutionary Design*. Springer, New York, NY, 2008.

[11] F.C. Hoppensteadt and C.S. Peskin. *Mathematics in medicine and the life sciences*, volume 10. Springer Science & Business Media, 2013.

[12] G.S. Hornby. Improving the scalability of generative representations for openended design. In R. Riolo, T. Soule, and B. Worzel, editors, *Genetic Programming Theory and Practice V*, Genetic and Evolutionary Computation Series, pages 125–142. Springer, Boston, MA, 2008.

[13] T. Kowaliw, N. Bredeche, and R. Doursat. *Growing Adaptive Machines*. Springer, 2014.

[14] C. Lee and D. Ashlock. Diffusion characters: Breaking the spectral barrier. In *Proceedings of the 2008 Canadian Conference on Electrical and Computer Engineering*, pages 847–850, 2008.

[15] Gilbert Syswerda. A study of reproduction in generational and steady state genetic algorithms. In *Foundations of Genetic Algorithms*, pages 94–101. Morgan Kaufmann, 1991.

[16] M. Timmins and D. Ashlock. Network induction for epidemic profiles with a novel representation. *Biosystems*, 162:205–214, 2017.