Data Visualization of Agent-Based Simulation of an Infectious Spread

Jingyi Gan

Tepper School of Business
Carnegie Mellon University
Pittsburgh, PA 15213
Email: kyragan@cmu.edu

Dominique Thiebaut

Dept. Computer Science
Smith College
Northampton, MA 01063
Email: dthiebaut@smith.edu

***Abstract*—We present a novel data visualization approach to display the contact trace of the spread of an infection between individuals. This visualization presents the spread as a radial organizational chart, where each node is an infected person, and the distance from the root to a node is proportional to time. We use real registration information for a population of students at a small college to generate a social network that is fed to an agent-based simulator. The simulation implements the Susceptible-Infected-Recovered (SIR) model to control how the infection moves from one individual to another. Contrarily to other models that generate expected quantities, our tool displays scenarios of a typical outbreak, where individuals involved in the spread are identified, along with the trace of their infection. The usefulness of our tool is in illustrating at the micro level phenomena such as the appearance of super-spreaders, or the influence of interventions such as quarantine or vaccination. We present several visualizations corresponding to different SIR parameters, and also illustrating the effect of vaccination.**

Keywords–data visualization; agent-based modeling; discrete event simulation; social network; contract graph; intervention strategies; contact-tracin; SIR model

I. INTRODUCTION

In this paper we present a novel approach for visualizing the *contact-trace* or *contact-map* [1] resulting from the spread of an infectious disease in a population whose *social network* [2], [3] is known a-priori. The contact map is generated from the data output by an *agent-based* simulator that uses the SIR model [4] to control agents representing the students enrolled at Smith College (2,625 students) during the fall semester of 2012, and for whom we have obtained the complete individual course registration (487 different courses), as well as their lodging information for that semester (49 different dorms). Because students at Smith College live on-campus, we can simulate their every-day contacts during class, during meals, and during study periods. The fine-grain simulation evolves on a one-hour scale, and lasts the 14 weeks of a semester, or shorter if the whole population gets infected.

Our contact-map is a variant of *radial organizational charts* [5], where a tree is displayed with its root in the middle of the graph, and all its descendants organized in a 360-degree fanout. In our implementation, each node of the tree is a student, and the root is the first infected student. Edges link students who directly infect other students, with the infecter closer to the root than the infected. New for this type of visualization, we set the length of an edge to be directly proportional to the time it takes for an infected student to infect another student. In the SIR model, individuals are infected only if they are susceptible, after which point they incubate the virus, and then become contagious for a given period of time, after which they recover and are not contagious any longer. Using time as the scale of the graph helps better understand how quickly the outbreak expands, and how long it lasts.

Different visualization attributes, such as node size and color, as well as edge width are available to enhance various properties of the infectius spread. We set the size of a node to vary proportionally to the number of other students directly and indirectly infected by the student associated with that node (number of tree descendants).

The advantage of such a map is that it can help the contact-tracing process at the beginning of an outbreak when a few individuals in a real population are found to be infected. If the social network of the whole population is known, then possible scenarios for the spread of the infectious disease can be plotted, and the efficacy of various preventive measures, e.g. vaccination, or quarantine, can be evaluated through simulation. The opportunity to assess visually the most probable path taken by an infectious disease as it spreads through a population is a beneficial complement to standard contact investigative techniques.

The population used in our simulation is a closed population, which we assume has no contact with the outside world. In a way we simulate a campus without staff or faculty. While it is a simplistic rendition of real life on campus, it does provide insights for important situations where the population is isolated from the rest of the world, such as in hospitals, on ships, or in small towns, as have been investigated in [6]–[8].

When studying an infectious outbreak, health researchers typically use stochastic models to assess the spread of a disease. Some tools use geographical information systems (see [9] for an example of a visualization of heat-maps of the spread of mosquito-based diseases), and others present statistical properties of the infected population over time. These tools provide a good understanding of the overall spread, but offer no knowledge of how the infection spreads from one individual to the next. The ability to trace a given individual and the spread of infection it creates, and to observe how key agents appear and affect the infection, such as *super-spreaders* [10], can help health professionals better control infection outbreaks.

In the next section, we review background information that puts our research in context. In Section II, we describe the real data we use to drive our agent-based model, which is presented in Section III. In Section IV, we present several data-visualizations illustrating how different SIR parameters

 ISBN: 978-1-61208-567-8

or interventions affect the contact map, and we provide an analysis of the graphs in Section V. Section VI concludes this paper.

A. Background

The major work that provides an overview of data visualization techniques in the health field is that of Carroll et al [11] who study a "myriad of new tools and algorithms [that] have been developed to help public health professionals analyse and visualize the complex data used in infectious disease control." Our visualization tool belongs in their *social network analysis* section, which they report as one of the most recent and growing fields of the health literature, accounting for approximately 10% of the total number of yearly health publications. While the general purpose of the tools surveyed is to address the identification of common characteristics, such as *risk stratification* of contacts, *identifying common characteristics* of those infected, *visually communicating* cases for improved understanding of outbreaks, and *identifying potential pathways of transmission*, they note that as network data becomes more available, new diverse methods of visualization will be needed [11]. We suggest our work fits in this arena.

Our work also parallels that of Hansen et al [6]. Their approach concentrates on visualizing possible scenarios of the spread of an infectious disease in a hospital, presenting the user with an interactive 3D graphic representation of the floors and rooms of a hospital, and how the infection spreads across the building. The data visualization is driven, as in our case, by an agent-based simulator which is fed the real-life records of the social contacts between health workers and patients. While their visualization has the added advantage of presenting the user with an interactive interface, it does not display a contact trace of the infection, although they very likely have access to the data needed to do so. We see our work as a logical extension, or addition to theirs.

Our data-visualization is also reminiscent of the *shortest-path tree* graph presented by Brokmann [12]. The nodes of Brokmann's tree are airports, and the edges are proportional to the geographical distance separating these airports. In contrast, the edges of our tree are proportional to time, and the nodes are infected individuals. In some ways Brokmann's edges represent an approximation of time, as well, since planes fly regularly from one airport to another, with approximately uniform speed. Our visualization allows an exploration on a much smaller scale than theirs, complementing their work as well.

In the next section, we present the social-network data used in our simulation.

II. Social-Network Data

Our data is taken from a spreadsheet maintained by the Registrar's office at Smith College which catalogs all classes for which all 2,625 students registered during the Fall of 2012. Each student is identified by a unique Id number. For each student, we have a record of the dorm she resides in and the courses for which she is registered. Each course has a unique Id. A student typically takes four courses a semester, and for each one we have available the daily/weekly time block(s) in which a course meets. The name of the buildings and the classroom numbers where the courses take place are also available. In addition, the type of classroom meeting is also recorded, e.g. studio, performance, lecture, lab, colloquium, discussion, or seminar. We do not use this particular information, but note that it could be used in future work to refine the granularity of the simulation, for example in quarantine scenarios. We process this data and create lists of Ids of students located in each classroom on campus in each time block of the week. These lists of Ids associated to location and time-blocks form the base of our social network. For social connections outside the classroom, we extrapolate the spreadsheet data and assume that students will take their three daily meals in the dorm in which they reside, and that they also study in their dorm after dinner and on weekends. In the next section we describe the discrete-event simulation that processes the list of Ids and controls the state of each agent as the simulated time passes.

III. The Agent-Based Model

The simulation keeps track of each student, or *agent*, during her weekly schedule, and maintains a status of her health according to the SIR model, where individuals evole through an epidemic by transitioning through different *states*. In the SIR model, somebody is initially assumed to be healthy and ***S**usceptible*, then gets ***I**nfected*, which results in an incubation period T_i during which the student is not contagious, followed by a period T_c where she becomes contagious, which finally ends with the student healing and switching to a ***R**ecovered* state. We assume that students maintain their regular activities while they are infected and contagious. In our model, we also allow for a (small) probability p_r for recovered students to remain contagious.

When a susceptible student enters a location where contagious students are located, she experiences a probability p to get infected by each one of them. Contagious students can be those who have recently been infected, or those who have recovered, but are still possibly lightly contagious (with probability p_r). In our SIR model, we assume that recovered students are imunized to future infection by the same virus.

The simulation lasts for a simulated time equivalent to a semester of 14 weeks, which matches exactly the duration of a Smith College semester. One student is picked at random (or not, if repeatable scenarios are of interest) at the beginning of the simulation, T_0, which coincides with the breakfast period of the first day of class. As the infected student goes about her daily schedule, she randomly infects the susceptible students who come in contact with her, in class, during meals, or during study periods.

The simulator is written in Java and takes an average of 7.8 seconds to run one simulation to completion on a 2.4 GHz Pentium Core i5 with 8 GB Ram. It generates a *contact-trace* of the spread of the infection as a collection of tuples of students Ids associated with a time. The time corresponds to the instant when the second student gets infected. These tuples form the edges of a tree data structure, which is recorded in *DOT* format [13], compatible with the *Graphviz* visualization package [14].

In the next section we present several data visualizations generated from the *dot* output of the simulator.

IV. Data Visualization

The graphs presented below are generated using *dot* (not to be confused with the dot language), one of the applications

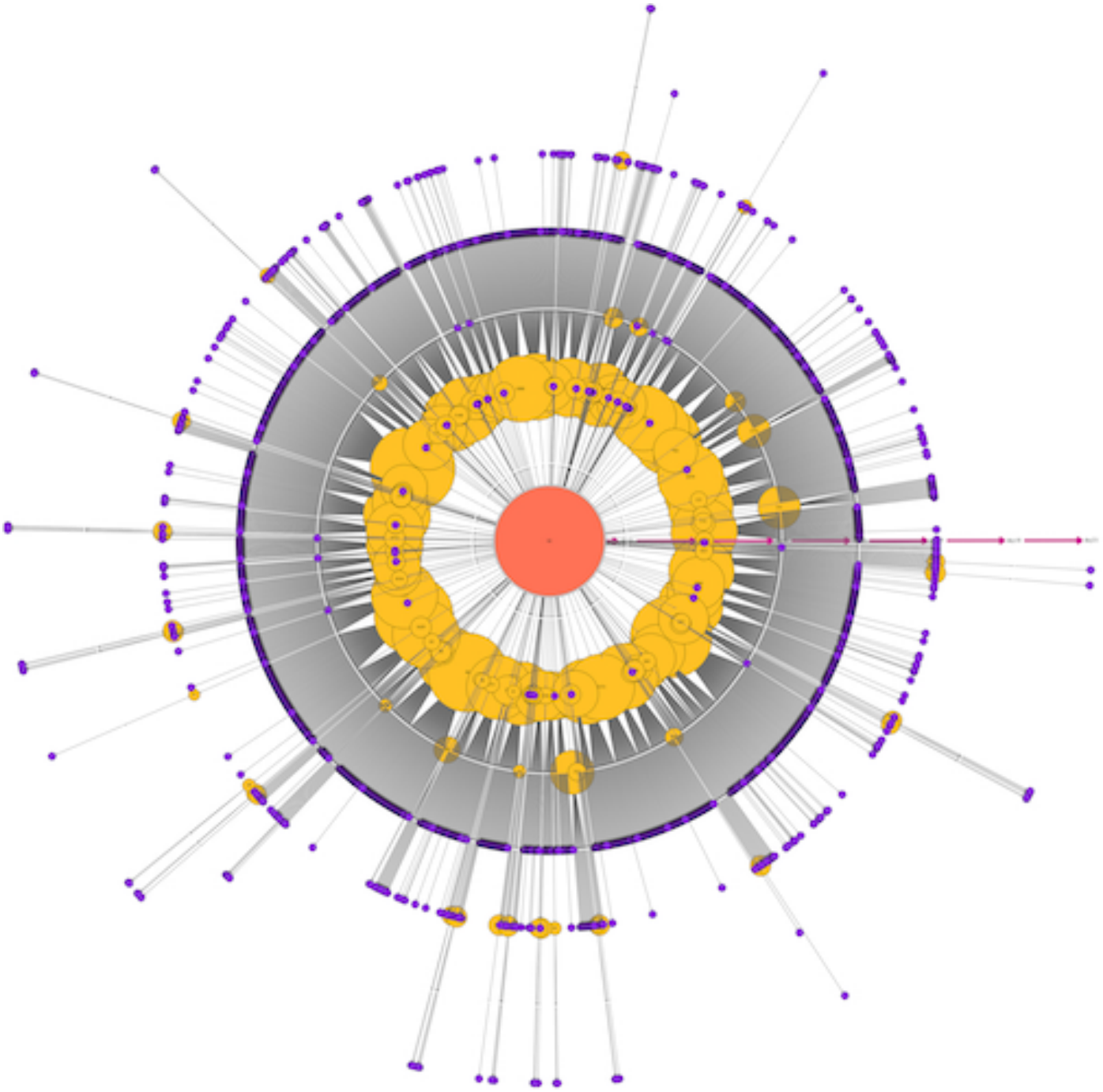

Figure 1. First infected is Student No. 82. T_i=6 days, T_c=8 days, no vaccination, no quarantine, p=1.0.

of the *Graphviz* package. Dot takes the dot-formatted file generated by the simulator and creates a graphic file of the resulting radial graph. We use the Scalable Vector Graphic [15] (SVG) format for the graphic output to fully capture the details of such a large number of tree nodes and levels. Generating the SVG file typically takes an average of 5.5 seconds on the same 2.4 GHz Pentium Core i5, and the resulting graphic file is 5 to 10 MBytes in size.

Figure 1 shows the contact-map resulting from a simulation where we set p=1.0, which ensures that if two students attend the same class, or share a meal in the same cafeteria, and one is infected, than the other automatically catches the infection. p=1 also ensures that the whole population gets infected (unless there exist subgroups of students who never interact with the larger population of students). While setting p to 1 is not a realistic situation, it presents the interesting boundary-case scenario that would result from an extraordinarily virulent infection. The parameters used in Figure 1 are p=1, p_r=0, T_c=8 days, and T_i=6 days.

Each graph also bares a time axis organized as a series of

arrows going from the center of the graph to the East, and complements the graph. In Figure 1, the length of each arrow corresponds to 3 days. The largest concentric circle has a radius of 21 days, indicating that the whole student population is infected after 21 days.

The first infected student is at the center of the graph, and is shown in red. The size of a node is proportional to the number of people infected by the student associated with that node. The radius of a node is defined as $radius = \log(1 + numberofdescendants) * 0.35$. Nodes other than the root are either orange of purple, depending on whether they infect several people, or just one, respectively. We note that most of the orange nodes fall on the first concentric circle of the circular graph, and are the largest of the tree, indicating that these students will behave as super-spreaders, as they have more time than the others to infect students they'll come in contact with. Because the time periods T_c and T_i are constant, and not taken from a distribution, all the nodes fall exactly on a few concentric circles, relative to the root.

Note also that the locations where the tree nodes are placed are algorithmically picked for optimal use of the space by the *dot* application, and slight variations in the tree may result in significantly different looking graphs.

Figure 2 shows a close-up region of Figure 1, illustrating the numbering of the nodes with the student Id, and the detail of the time scale.

When p is set to a more realistic value of 0.01, we obtain the graph depicted in Figure 3. It now takes 36 simulated days for the whole population to be infected, but the dynamics at the beginning of the infection is more complex than observed in Figure 1, with a distribution of differently sized super-spreaders who start their infectious path around Days 15 and 18.

In Figure4, we show how the visualization can help health officials understand the effect of various interventions. In this graph we assume that 50% of the initial population of students is vaccinated at the beginning of the semester, and that those vaccinated have a 0-probability of getting infected, or of becoming carriers of the infection. Since the visualization only shows infected students, Figure 4 contains only half of the population of students, namely those not vaccinated. The outbreak lasts 55 days, and a total of 15% of the total student population gets infected during this time, or 30% of the non-vaccinated students. Here again, we have very different dynamics at play, with a handful of super-spreaders who propagate most of the infection; they are the orange nodes appearing between the Day 24 and Day 36.

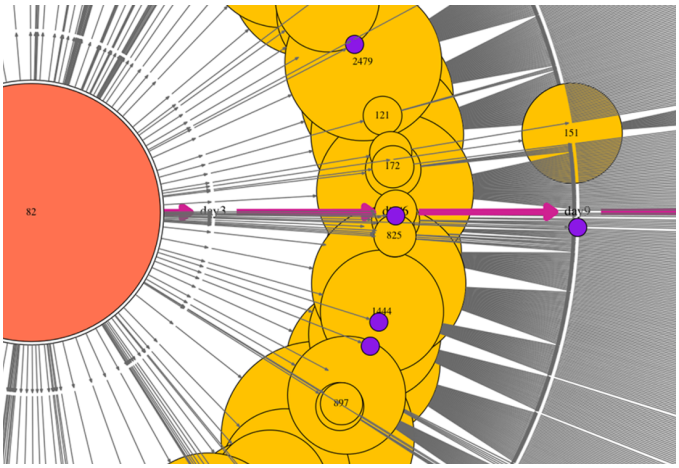

Figure 2. Close-up of Figure 1, showing details including node labeling and time scale.

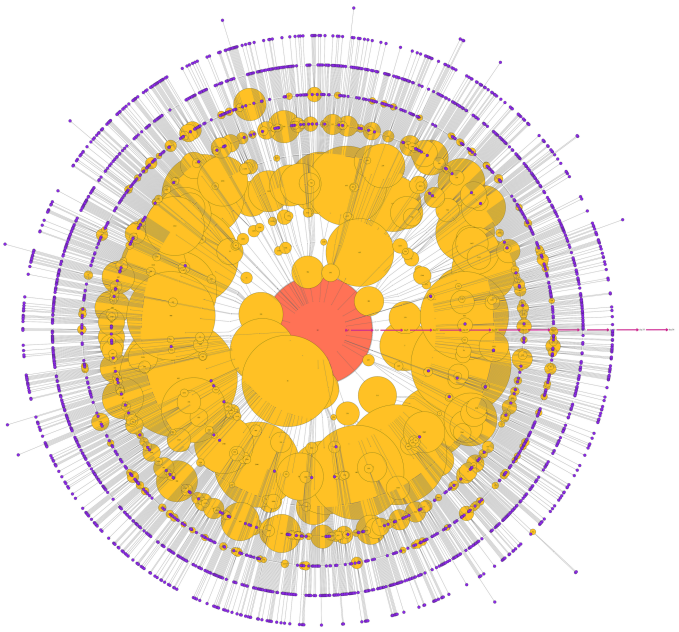

Figure 3. Contact map for p=0.01, T_i=6 days, and T_c=8 days.

V. ANALYSIS

Our model and visualization present new insights in the way an infectious disease spreads in a closed population for which the social network is well defined. Each figure represents one of many possible scenarios, and should not be seen as an average behavior; just a probable one. Unless the seed of the random number generated remains the same for different simulations, two different simulations with the same initial parameters and root student will yield two different trees. Whether the simulated growth of the infected population bears a chaotic component is open for research, however, it is helpful to see the trees generated by the agent-based simulator as different expressions of some dynamical system, all with the same *strange attractor* [16]. Our data visualizations present the micro-level dynamics of the infection, rather than an average variation of some quantity.

It is easy to see that given an infected student in the population, our model provides an exact trace of who infects her, and who she infects in turn. Moreover, the day and location of the infection from one student to the other is known exactly. Such information could easily be added to an interactive version of our visualizations.

Our visualizations also offer the ability for health officials to investigate an infectious spread in its early stage, when just a few students are found to be infected. Assuming the social network for the population is available, a modified data visualization can show the group of infected students in a collective multi-node root of the tree, and the trace of

 ISBN: 978-1-61208-567-8

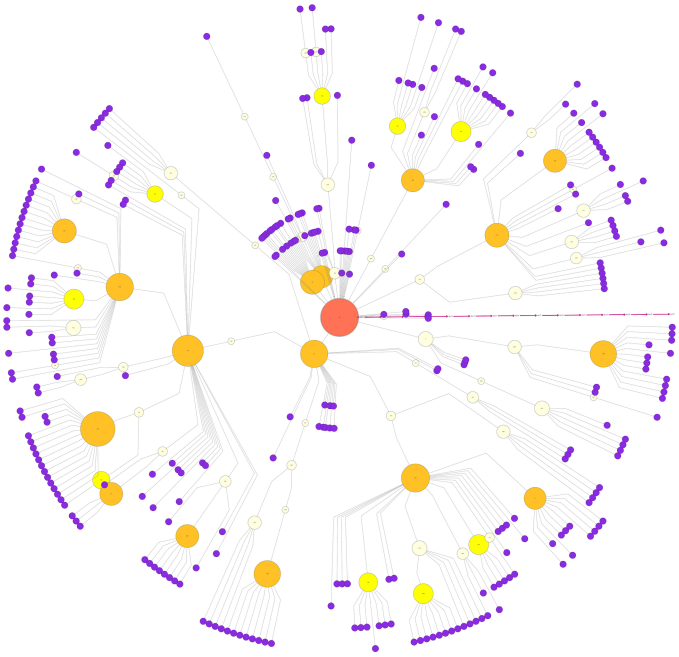

Figure 4. Contact-map for p=0.01, T_i=6 days, and T_c=8 days, with 50% of the initial population vaccinated. The probability that a vaccinated person is contagious is 0.

potential contacts emanating from it. Officials can then use this information to order local quarantines on buildings or dorms, or cancellation of meetings in particular locations or time blocks.

Different visualization attributes, such as node size and color, as well as edge width are available for enhancing various properties of the infectious spread. We decided to use the size of a node to grow proportionally to the number of other students directly and indirectly infected by its associated student. Interaction and animation could also enhance the visualization; time-lapse growth of the tree, or selection of particular branches or nodes, for example, could enhance the usefulness of our tool. We note, however that the size of the population makes it challenging to display the entirety of the tree with good resolution.

VI. Conclusions

In [11], Carroll et al. review visualization and analytical tools for infectious disease, and state "visualization methods to help users understand network structures have not been widely employed in tools for public health." Our visualization tool answers this call and presents a novel approach for evaluating probable spreads of an infectious disease in a closed population with a known social network.

The radial organization provides a low-level understanding the dynamics of an infection, and how different parameters such as vaccination, or quarantine, can affect its spread, as illustrated in Figure 4.

Several improvements to the model are possible. For example, we could take T_i and T_c from a distribution other than the uniform distribution. We could also create super-spreaders by picking several agents before or during the simulation, and by giving them an a-priori probability distribution for the virulence with which they act. Both the population size of super-spreaders and their virulence can easily be coded in the model. The model can also be augmented so that various scenarios are triggered automatically when a particular threshold of infection is detected in the simulation. Such scenarios could involve the cancellation of classes taking place in amphitheatres, or forcing students to eat their meals in their dorm room.

We also noted earlier that an interactive visualization could provide additional information that is available but impossible to display on a static image. This include offering the user an interactive menu to modify key SIR and visual parameters, as is presented in [17]. Other improvements include generating a full contact path between selected students showing the identity of the students, the location and time of the contacts.

Finally, we note that our visualization tool could be used to evaluate various properties of key agents, such as super-spreaders, and compare simulation outputs to real data in an effort to find the model parameters best matching observed behavior. We have however to take Carroll's advice seriously, when he and his coauthors state [11] that visualization tools also risk misleading users due to misinterpretation or cognitive overload. Sometimes, simpler is better.

The code for the agent-based simulator can be found on this repository [18].

Acknowledgments

We gratefully acknowledge the insightful comments and suggestions given to us by Professors Sarah Moore, of the engineering department, and Robert Dorit, of the department of biological sciences, both at Smith College.

References

[1] S. Etkind, "Contact tracing. TB: a comprehensive international approach," in Lung Biology in Health and Disease, L. Reichman and E. Hershfield, Eds. New York, NY: Marcel Dekker, 1993, pp. 275 – 289.

[2] F. Viegas and J. Donath, "Social network visualization: Can we go beyond the graph," in Workshop on Social Networks, CSCW, vol. 4, 2004, pp. 6–10.

[3] P. McElroy, R. Rothenberg, R. Varghese, R. Woodruff, G. Minns, S. Muth, L. Lambert, and R. Ridzon, "A network-informed approach to investigating a tuberculosis outbreak: implications for enhancing contact investigations," The International Journal of Tuberculosis and Lung Disease, vol. 7, no. Supplement 3, 2003, pp. S486–S493.

[4] R. M. Anderson and R. M. May, Infectious Diseases of Humans: Dynamics and Control. Oxford University Press, 1992.

[5] R. L. Harris, Information Graphics: A Comprehensive Illustrated Reference. Oxford University Press, 2000.

[6] T. E. Hansen, J. P. Hourcade, A. Segre, C. Hlady, P. Polgreen, and C. Wyman, "Interactive visualization of hospital contact network data on multi-touch displays," in Proceedings of the 3rd Mexican Workshop on Human Computer Interaction, ser. MexIHC '10. San Luis Potosi, S.L.P. Mexico, Mexico: Universidad Politecnica de San Luis Potosi, 2010, pp. 15–22. [Online]. Available: http://dl.acm.org/citation.cfm?id=1978702.1978708

[7] B. Yu, J. Wang, M. McGowan, and G. Vaidyanathan, "Agent-based stochastic simulations of shipboard disease outbreaks," in Proceedings of the 2010 Spring Simulation Multiconference, ser. SpringSim '10. San Diego, CA, USA: Society for Computer Simulation International, 2010, pp. 123:1–123:8. [Online]. Available: http://dx.doi.org/10.1145/1878537.1878666

[8] J. B. B. C. D. Shamir, N. M. M. Laskowski, and M. F. R. D. McLeod, "Smartphone technologies for social network data generation and infectious disease modeling," Journal of Medical and Biological Engineering, vol. 32, no. 4, 2012, pp. 235–244.

 ISBN: 978-1-61208-567-8

[9] S. M. Mniszewski, C. A. Manore, C. Bryan, S. Y. Del Valle, and D. Roberts, "Towards a hybrid agent-based model for mosquito borne disease," in Proceedings of the 2014 Summer Simulation Multiconference, ser. SummerSim '14. San Diego, CA, USA: Society for Computer Simulation International, 2014, pp. 10:1–10:8. [Online]. Available: http://dl.acm.org/citation.cfm?id=2685617.2685627

[10] W. Duan, X. Qiu, Z. Cao, X. Zheng, K. Cui et al., "Heterogeneous and stochastic agent-based models for analyzing infectious diseases' super spreaders." IEEE Intelligent Systems, July/August 2013, pp. 18–25.

[11] L. N. Carroll, A. P. Au, L. T. Detwiler, T. chieh Fu, I. S. Painter, and N. F. Abernethy, "Visualization and analytics tools for infectious disease epidemiology: A systematic review," Journal of Biomedical Informatics, vol. 51, 2014, pp. 287 – 298. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1532046414000914

[12] D. Brockmann and D. Helbing, "The Hidden Geometry of Complex, Network-Driven Contagion Phenomena," Science, vol. 342, no. 6164, Dec. 2013, pp. 1337–1342. [Online]. Available: http://dx.doi.org/10.1126/science.1245200

[13] E. Gansner, E. Koutsofios, and S. North. Drawing graphs with dot. http://graphviz.org/Documentation/dotguide.pdf. (Accessed Nov. 1, 2006)

[14] A. Bilgin, J. Ellson, E. Gansner, Y. Hu, and S. North. Graphviz - graph visualization software. http://graphviz.org/Documentation/dotguide.pdf. (Accessed Nov. 1, 2016)

[15] O. Andersson, P. Armstrong et al. W3c working draft of scalable vector graphics 1.2. http://www.w3.org/TR/SVG12/. (Accessed Nov. 1, 2017)

[16] F. Takens, Detecting strange attractors in turbulence. Berlin, Heidelberg: Springer Berlin Heidelberg, 1981, pp. 366–381. [Online]. Available: http://dx.doi.org/10.1007/BFb0091924

[17] W. V. d. Broeck, C. Gioannini, B. Gonçalves, M. Quaggiotto, V. Colizza, and A. Vespignani, "The gleamviz computational tool, a publicly available software to explore realistic epidemic spreading scenarios at the global scale," BMC Infectious Diseases, vol. 11, no. 1, 2011, p. 37. [Online]. Available: http://dx.doi.org/10.1186/1471-2334-11-37

[18] J. Gan. Visualizing the transit map of the spread of an infectious disease. http://tinyurl.com/kyra-is. (Accessed Nov. 1, 2016)

 ISBN: 978-1-61208-567-8