Investigating the Effectiveness of Wastewater Surveillance Through Computer Simulation

A Technical Report submitted to the Department of Computer Science

Presented to the Faculty of the School of Engineering and Applied Science

University of Virginia • Charlottesville, Virginia

In Partial Fulfillment of the Requirements for the Degree

Bachelor of Science, School of Engineering

Colin Crowe

Fall, 2022

On my honor as a University Student, I have neither given nor received unauthorized aid on this assignment as defined by the Honor Guidelines for Thesis-Related Assignments

Nathan Brunelle, Department of Computer Science

Investigating the Effectiveness of Wastewater Surveillance Through Computer Simulation

Colin Crowe
A paper submitted to the University of Virginia

1 Abstract

Wastewater surveillance helps monitor infectious disease outbreaks by collecting samples from sewage and testing them for presence of the disease. Because each wastewater sample contains waste from many individuals, wastewater surveillance generally cannot be used to effectively identify specific infected individuals and issue quarantine guidance. To investigate the degree to which epidemiologists can mitigate this limitation, this work presents a computer simulation of disease spread and corresponding flow of infected wastewater. This computer simulation consists of two layers. The top layer operates as an agent-based SIR (Susceptible, Infected, Recovered) model in which nodes could infect each other, and the bottom layer represents sampleable sewer manholes that may contain the disease based on the activity on the top layer. This simulation was then run on two different graphs, each with different densities of interconnectivity among the nodes on the top layer. The data produced by our simulations suggests that the lower interconnectivity in the SIR graph enables more effective wastewater surveillance, implying that real-world isolation and quarantine makes for more effective disease monitoring.

2 Introduction

The onset of the COVID-19 pandemic motivated developing new techniques in monitoring disease spread. One such technique that gained prominence during the pandemic is wastewater surveillance[1]. In essence, if someone is infected, then traces of COVID can be found in the sewage they produce. Wastewater surveillance leverages this to monitor the spread of disease by taking samples of sewer water, testing them for traces of COVID, and using the test results to locate infected areas.

However, wastewater surveillance possesses limitations which make it an imprecise tool. The effectiveness of wastewater surveillance as a method for monitoring disease depends on our ability to derive the origin of each infected sample, but there are myriad issues that can arise in this process. In addition to unavoidable issues, such as sample transportation time, sample storage, sewage system leaks, etc., there are also several logistical issues unique to wastewater monitoring, such as sampling location, frequency of sample collection, and homogenization of the sewage flow, all of which can make this process more difficult[1]. The cumulative result is that, even if a positive trace is detected, these limitations make it difficult to identify any specific individuals or groups who need to take action.

One final complication with wastewater surveillance is that its application lies in disease control, a field already subject to quite a lot of uncertainty. It is difficult to confirm the results produced by wastewater surveillance because this requires testing real people for real

disease. It's no surprise that, rather than try and conduct a real-world experiment, epidemiologists instead turn to computer simulations to answer such questions, which have the distinct advantage of allowing the modeler to see any data about the situation they might need. In this work we explore whether such a technique could be used to improve wastewater surveillance.

Some issues, like sewage leaks, are difficult to account for. However, other issues, such as sampling location and frequency of collection, can be accounted for through a more intelligent process for conducting wastewater surveillance. The purpose of this research is therefore to investigate the circumstances that make wastewater surveillance effective or ineffective through the use of a computer simulation and to gain insight into how good circumstances can be created and the effectiveness of disease control improved.

3 Related Work

A computer model seeking to investigate wastewater surveillance must both simulate how infection spreads throughout a community as well as how that community affects the sewer network beneath it. We leverage prior work to account for both factors.

3.1 Epidemiological Models

Examples of epidemiological applications of computer simulations are plentiful. The University of Virginia's very own Biocomplexity Institute, for example, makes use of computer models to inform local and federal governments about the state of the COVID-19 pandemic [2].

Members of the Biocomplexity Institute have worked on several relevant studies, but one in particular seems similar to our problem. In collaboration with a team at the University of Maryland, they investigated how contact tracing can be made more effective to similar ends: more effective contact tracing, just like more effective wastewater surveillance, reduces the burden of isolation by identifying only a few people who need to quarantine rather than the whole population. They constructed a graph consisting of nodes which can be susceptible, infected, or recovered (SIR), and edges that connect them to other nodes, thus allowing the disease to spread. They then developed algorithms to determine who needs to isolate based on the results of the simulation[3].

The above study could serve as a blueprint for this investigation, but that does not mean alternatives should not be explored. For instance, that study used an "agent-based" approach, in which the model consists of individual agents (nodes) who interact with each other over the course of the simulation. This is as opposed to a "compartmental" model, in which, rather than have distinct agents, the model consists of a number of compartments which represent the populations in each category[4]. In an SIR model, there would be a compartment for all three of susceptible, infected, and recovered that represents the number of people in that category. As the simulation runs, those populations change to reflect people getting infected or recovering from the disease, rather than representing that by encoding every individual as an agent in a graph.

Additionally, there is the possibility of changing the number of compartments. The aforementioned study used only three: susceptible, infected, and recovered, but those categories can be split up more than that. An example would be using an SEIR model, with compartments for susceptible, exposed, infected, and recovered nodes[4]. The advantage here is to simulate an incubation period for the disease. In the real world, it is possible for someone to be contagious but not realize it, and to continue to spread the disease before their symptoms develop and they realize they have been infected. The SIR model does not give room for such behavior, while an SEIR model would.

3.2 Sewer System Models

While there were plentiful examples of epidemiological models to pull from, we found relatively few examples of computer simulations being used to better understand sewer systems. Even still, that does not mean such studies do not exist. In fact, one study seemed especially similar to our own problem.

Chemicals like sulfuric acid, sodium hydroxide, and sodium sulfate can be dangerous if discharged into sewage lines, especially as those chemicals reach wastewater treatment plants. However, because these chemicals can get diluted the longer they stay in the sewer lines, it is necessary to monitor as closely as possible to the chemicals' source in order to track unlawful disposal of these harmful chemicals. Surveillance that extensive is difficult to maintain, though, so a team of researchers sought to investigate how this surveillance could be made more effective through computer simulation[5].

This solution involves deploying an Internet of Things approach (IoT), essentially making use of several small interconnected devices to monitor the sewer network. Conversely, our investigation does not seek to use an IoT approach, but instead seeks to leverage pre-existing sample locations, making this study a poor fit for emulation despite facing a similar problem.

Another related study sought to identify how heat from sewage could be used as a source of energy. While that seems quite different from the problems being tackled by this paper, modeling the thermodynamics of sewage did require developing equations to model how that sewage moved. They identified three factors important enough to parameterize when calculating wastewater movement: water from upstream, rainwater, and runoff from buildings[6]. They developed equations to calculate how each of these three factors would affect movement. As my investigation also needs to encode the movement of wastewater, those three equations could be useful here as well.

4 Model Design

There are always myriad decisions to be made when creating computer models, each with their own set of tradeoffs. This section consists of two sub-sections, one describing the model as it ended up, and the other justifying the choices made during the creation process.

4.1 The Model

At a high-level, the model consists of two layers. On the "top" layer is an agent-based epidemiological model of disease spreading throughout a community. On the "bottom" layer is a simple node-based representation of the sewer network beneath that community. The simulation runs for a set number of time steps, during which nodes can interact with each other and samples in the sewer can propagate.

The top layer is an undirected graph which consists of nodes that can be in one of three states: either they are susceptible, infected, or recovered (SIR). Susceptible nodes can become infected if there is an edge connecting them to an infected node. The probability that a node becomes infected during any given time step depends both on the baseline infection rate, given as input to the model, as well as the weight of the edge connecting it to an infected node, where a higher weight means a higher probability of infection. If a node is infected, there is a chance for it to become recovered during each time step. The probability for a node to become recovered is fixed and is also given as input to the model. Once a node becomes recovered, there is no mechanism to make it either susceptible or infected again, essentially assuming recovered individuals are immune from the disease.

The bottom layer is a directed graph of sewer nodes representing manholes. These manholes have a "status" variable that controls whether or not a sample collected from them will contain traces of disease. Each node in the top layer is connected to exactly one manhole in the sewer layer, and the weight of the edge connecting them determines how long it takes for a sample to travel from the node to the manhole. For instance, if the weight of an edge connecting a node to a manhole is 3, then, if that node becomes infected, it will take three time steps for the manhole's status to change as a result of that infection. The status of each manhole is updated every time step, and manholes do not have "memory" of their past statuses. In other words, if a manhole does not receive any infected samples during a time step, its status will read as not infected, even if it previously contained infected samples.

Each manhole, with one exception, feeds into another manhole. Edges connecting manholes to manholes work in exactly the same way as edges connecting nodes to manholes. The weight of the edge connecting them determines how long it takes an infected sample to travel to that manhole, and the status of that manhole is updated as if it were any other sample. Two manholes cannot both feed into each other. These rules ensure that there is always only one path for samples to take starting from any given node. The manhole labeled "end" is a special manhole added automatically to the end of the path. This is the only manhole that does not feed into anything else, and is analogous to a sewage treatment plant. The "end" node is the only way infected samples are removed from the network; they do not deteriorate on their own, so they stick around until they reach the end of the network.

A simple example network is shown in Figure 1 below. Nodes in blue are in the top SIR layer, while nodes in orange are in the bottom sewer layer. For example, if the node labeled "5" gets infected, then it could infect the nodes labeled "4" and/or "6", as they share an edge with 5. Samples from node 5 feed into the manhole labeled 4-5. Samples from that manhole will reach 6-7-8-9-10 once 6 time steps have passed and, eventually, enter the node labeled "end" after 4 time steps. Samples from node 4-5 do not flow into node 0-1-2-3; sewer samples only propagate in the direction of the end node. Note that, while every edge has a weight in

this network, only edges connecting sewer nodes have edge weights displayed for readability purposes.

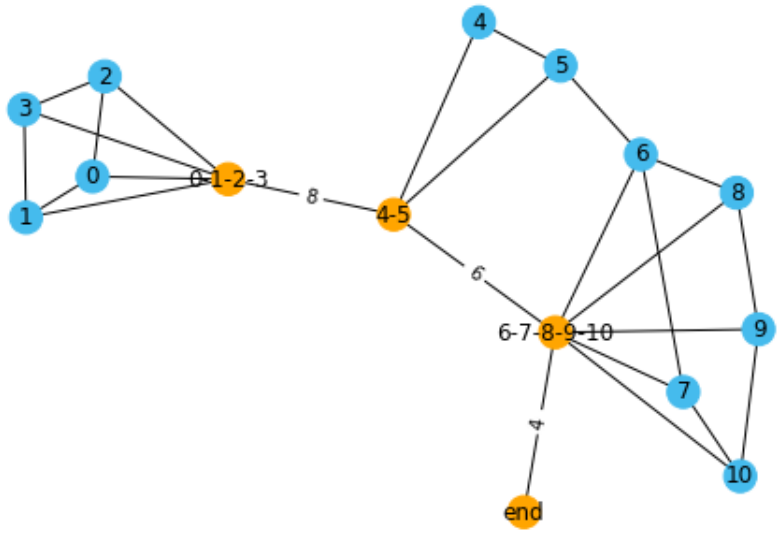

Fig. 1: Simple toy network containing 11 SIR nodes and 4 sewer nodes

When the simulation starts, a number of nodes are randomly selected to be infected (analogous to being "patient zero"). The number of nodes infected this way is specified by input, and this is the only way nodes can be infected besides infecting other nodes. While the simulation is running, the input parameters are fixed; parameters such as the base infection rate, recovery rate, and the shape of the network, including the edges and their weights, cannot be changed.

This section is only a summary of the model's functionality, however. To see the implementation in its entirety, see Appendix A.

4.2 Tradeoffs

It is important to acknowledge the choices made when developing computer simulations and how those choices may affect the results of that simulation. This section is therefore devoted to exploring alternative designs, moving from the broad high-level model functions and descending down into small details.

First, both layers of the graph are agent-based rather than categorical. This was done because the physical positions and locations are very important to this investigation. We need to be able to say that, for example, "X household feeds into Y sewer node", and making a statement like that becomes much harder without some sort of graph to model it. Choosing an agent-based model was therefore the obvious choice, even if it leads to a more computationally intensive model.

Second, both the SIR network and the sewer network are unchanging during the simulation. An assumption was made that, in the event of a pandemic, who individuals make contact with and how often they make contact would not change much over the course of the disease. In

other words, an individual either respects stay-at-home orders, or they do not, and their level of compliance does not change over time.

This is a more contentious decision. While the above assumption does seem reasonable, an equally reasonable assumption would be that individuals who are infected would make more of an effort to stay at home than individuals who are not. Implementing around this assumption would likely have resulted in splitting the infected state into two categories: exposed and infected, resulting in an SEIR model instead of an SIR model. Under such conditions, nodes that are infected would have their edge weights drastically reduced to represent self-isolation, while nodes who are exposed keep their original edges. This path was considered during model creation, but ultimately we decided to keep with the SIR model since, as the literature review section showed, SIR models can still be used to give good results. Adding this extra functionality just seemed unnecessary.

Third, there is no mechanism that allows nodes to transition from being recovered back to being susceptible. An alternative idea was explored early on, in which all recovered nodes become susceptible again at random intervals to simulate a new variant of the disease, but this was ultimately scrapped. Simply simulating the initial outbreak was found to give enough information to be useful, rather than simulating an elongated pandemic scenario.

Fourth, the sewer system doesn't make use of any equations presented in [6] to simulate the movement of wastewater. Instead, the weights on edges connecting manholes simply encode how many time steps it takes for samples to reach that node. This was done because, aside from rainfall, the factors identified that contribute to how long it takes for samples to reach different manholes don't seem to fluctuate very much. It was therefore assumed that wastewater flow could be approximated using a simple mechanism like this rather than trying to implement more complicated equations.

Fifth and finally, the "patient zero" of the simulation is chosen randomly each time. At first, the initial infected was specified before the simulation ran, but it was deemed that this made it unclear whether different results would be observed if a different patient zero was chosen. Conducting the simulations with a random patient zero made it so there was no dependency between the observed results and which node was chosen to be infected first.

5 Experiments

The basic experimental design for this model is to create a graph exhibiting some property and conduct several simulations on that graph to determine what patterns emerge. Ultimately, only one graph was designed for this purpose, as it was found that just one graph displayed enough interesting phenomena to make for good discussion. That graph is shown on the next page in Figure 2, with all edge weights visible. The input parameters that this graph was generated from are also included in the repository at Appendix A.

That graph consists of three different communities that together form a "three-pronged fork" in the sewer network. Branches in this network would be nodes 0-1-2, 8-9, and 14-15-16-17, while junctions would be nodes 4-6-7 and sewer node 25. The graph was not modeled after any real-world community; rather, most edge weights and node positions were chosen

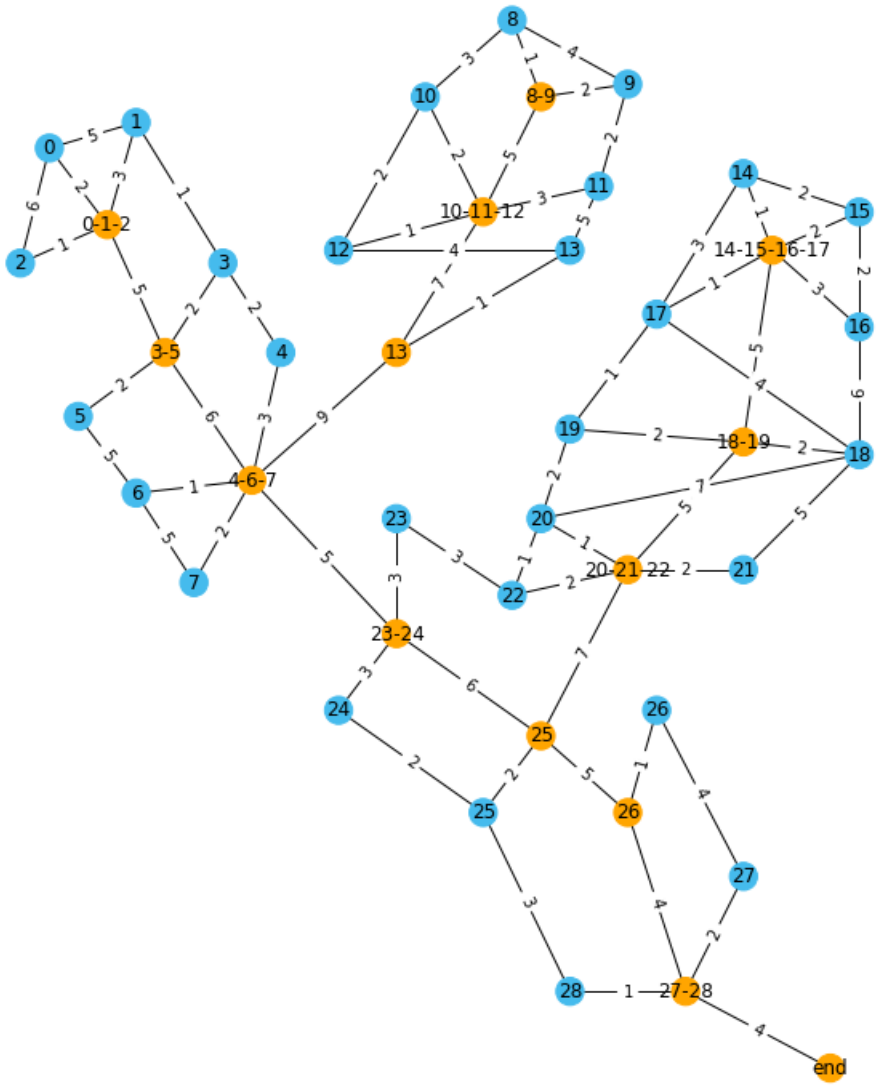

Fig. 2: Base graph used for experimentation

relatively arbitrarily. The only constraints were to be mindful not to make edge weights absurdly large, and that edges connecting sewer nodes to sewer nodes should typically have a larger weight than edges connecting SIR nodes to sewer nodes. The end result is that this network should be a reasonable approximation for a hypothetical community rather than a direct one-to-one mapping.

In order to run the experiments, a second variant of this graph was created with many more edges. This way the two graphs can be compared, with the above graph corresponding to a relatively isolated scenario in which different communities do not interact and cannot easily spread the disease to each other, and the below graph corresponding to a relatively interconnected scenario in which different communities are in frequent contact. This second graph is shown in Figure 3 on the next page. Note that, while every edge in the graph has a weight, only edges connecting sewer nodes have their weights displayed for the purposes of readabil-

ity. To see the exact parameters used, look again to the repository in Appednix A.

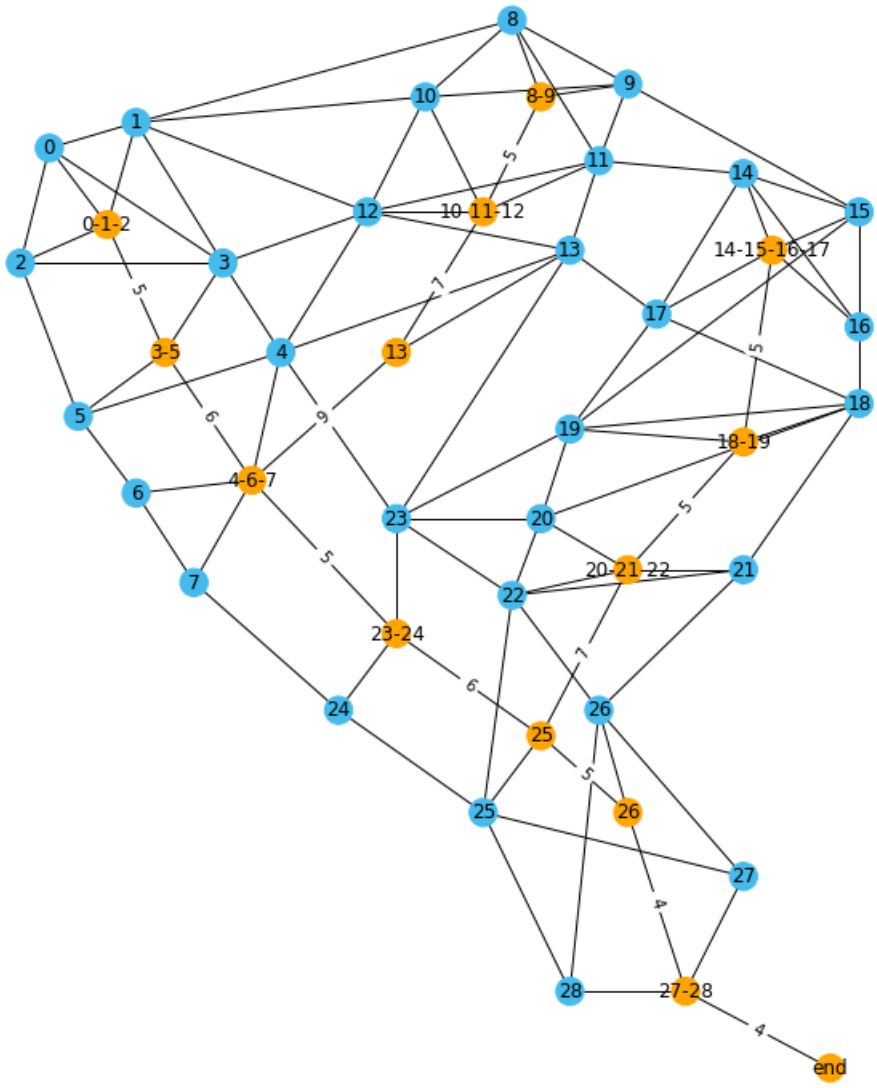

Fig. 3: “Interconnected” variant of the previous graph

Outbreaks of disease were then simulated on these graphs $1,000$ times each. Each simulation lasted for 50 time steps, and started with 4 nodes being initially infected. Each simulation was conducted independently of each other; information from the previous run does not carry over into the next one. During each time step of each run of the simulation, the statuses of every SIR node and every sewer node was recorded in a spreadsheet. Every experiment therefore resulted in a data set containing 50,000 rows, one for each time step of each simulation, and 43 columns, one for each node. This readout of the infection history was then analyzed to get the results.

6 Results

The first question to answer was to determine how a manhole containing a positive versus negative sample affected the probability any given node was infected. To this end, we calculated two probabilities per node per graph: the base infection rate, or the probability that a node was infected at any given time step, and the conditional infection rate, or the probability that a node was infected given that a specific sewer node contains a positive sample. Doing this analysis on the isolated model with sewer node 0-1-2 as the given sewer node, we observe the results shown in Figure 4 below, where the dashed line represents the average base infection rate:

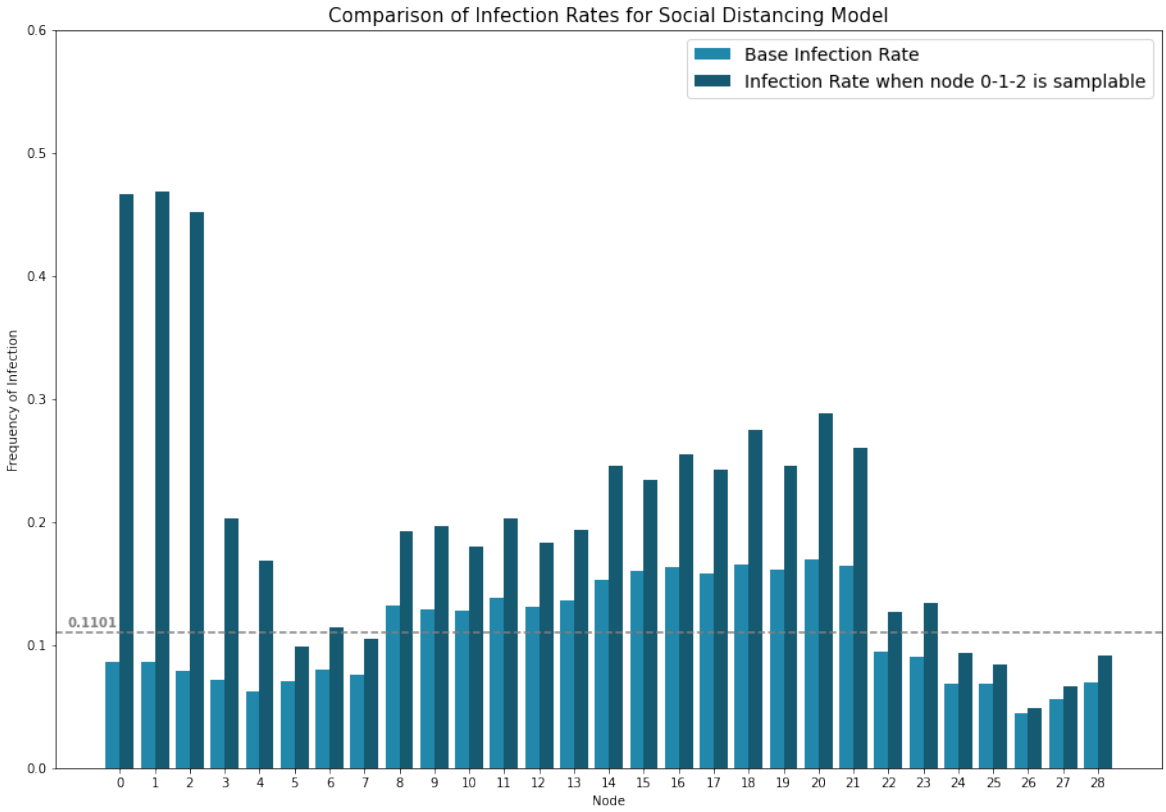

Fig. 4: Infection rates on the isolated model, looking at sewer node 0-1-2

The same analysis was also done on the interconnected graph, shown in Figure 5.

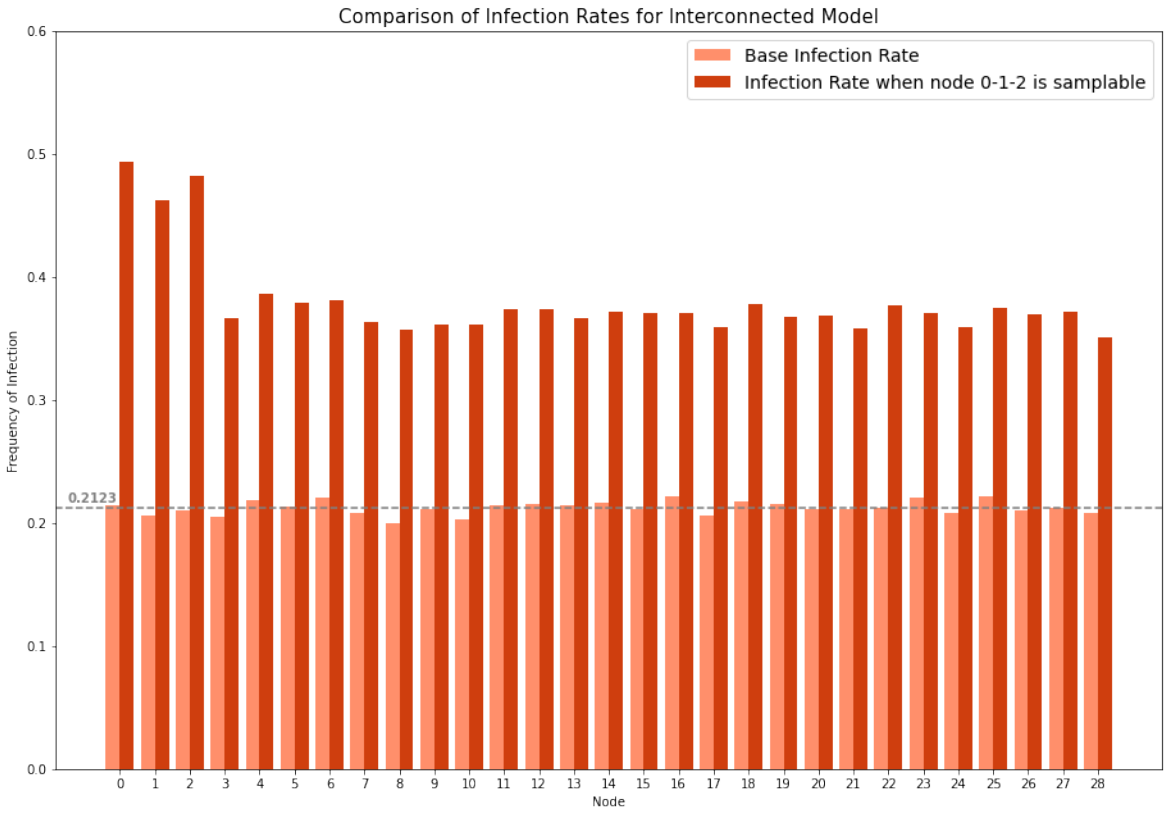

Fig. 5: Infection rates on the interconnected model, looking at sewer node 0-1-2

At first glance, it is difficult to draw any conclusions from these graphs because of how different their baselines are. The isolated model had a base infection rate that is roughly half that of the interconnected model, and so it is difficult to say if the differences we observe are a direct result of the additional edges or if it's simply that a higher infection rate led to these differences. Granted, it can be said that the extra edges led to a more infectious simulation, but it can also be said that a more infectious disease could do the same thing. In order to say that interconnection or isolation makes wastewater surveillance more or less effective, we need to make both scenarios display the same baseline.

To resolve this, two more experiments were conducted. The previous two experiments used the same input parameters both times, but, as these results show, doing so makes it difficult to compare the outcomes. So, the recovery rate for each graph was skewed in order to produce a higher base infection rate on the isolated graph and a lower base infection rate on the interconnected graph. The results of these are shown in Figure 6 below:

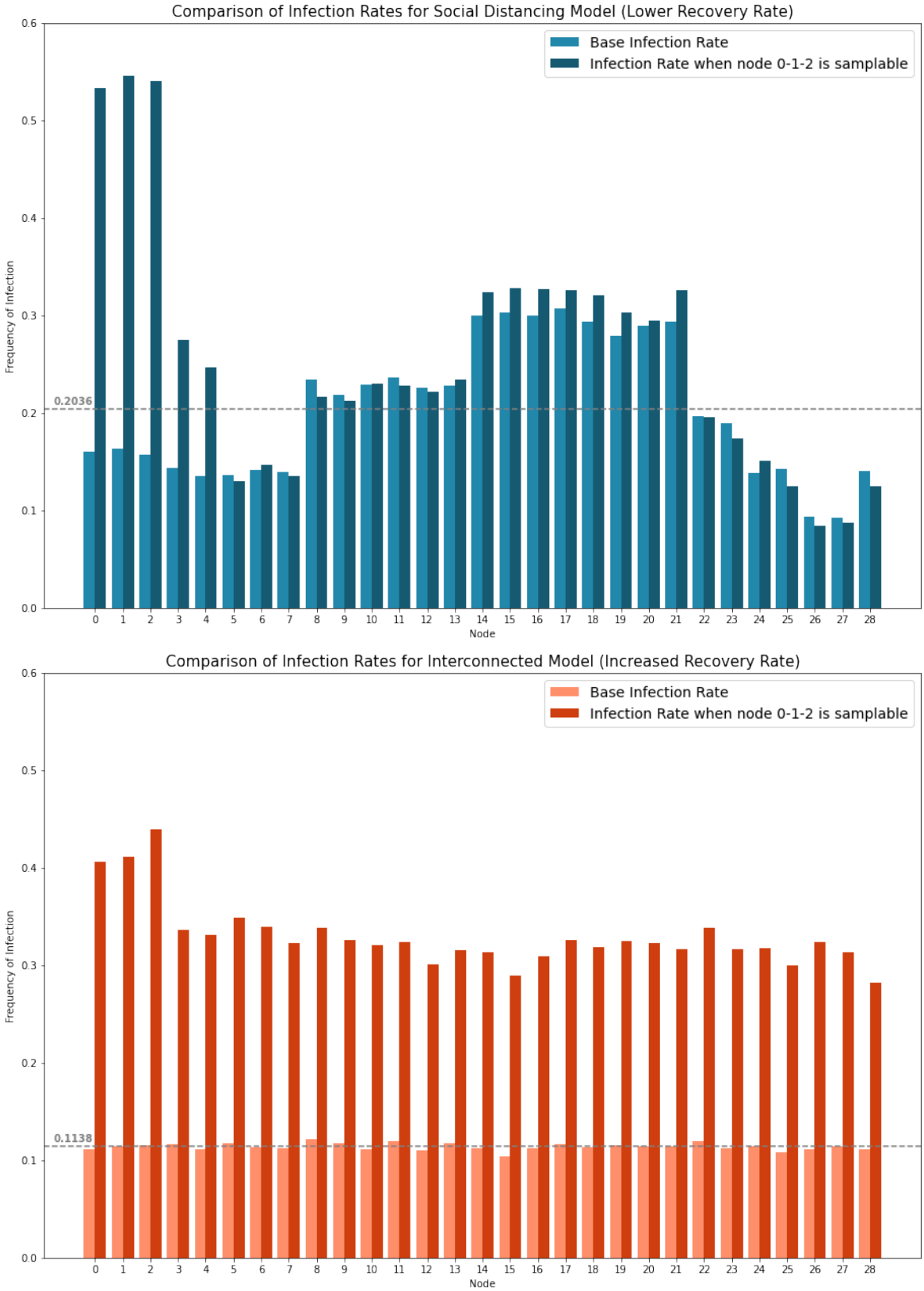

Fig. 6: Infection rates on the updated models. The isolated graph (top) now displays a base infection rate of approximately 0.2, while the interconnected graph (bottom) now shows a base infection rate of approximately 0.1

As can be observed, we now have a data set on the isolated model whose base infection rate matches the previous trial on the interconnected model, and we also have a data set on the interconnected model whose base infection rate matches the isolated model. In both cases, the overall behavior of the model appears to have been preserved.

For the purposes of the analysis moving forward, the data sets with base infection rates of approximately 0.2 will be used over the ones with base infection rates of 0.1. This means that simulations that led to the first interconnected graph and the second isolated graph will be the focus of our analysis. Note that this decision is relatively arbitrary and that the data sets with infection rates of 0.1 could have been used without significantly changing the conclusions.

Looking at the two graphs, they seem to support the idea that fewer interactions leads to more effective wastewater surveillance. In the isolated model, the probability that any given node is infected slightly increases for some nodes if we know that manhole 0-1-2 contains a positive sample, but only a few nodes show a dramatic increase in probability. In contrast, in the interconnected model, every node shows a high increase in probability if a positive sample is observed in manhole 0-1-2.

In fact, we can put a number to just how much the difference is. Measuring the covariance between node infection and manhole 0-1-2 containing a positive sample allows us to quantify how much these variables vary together. The result of this is shown in Figure 7:

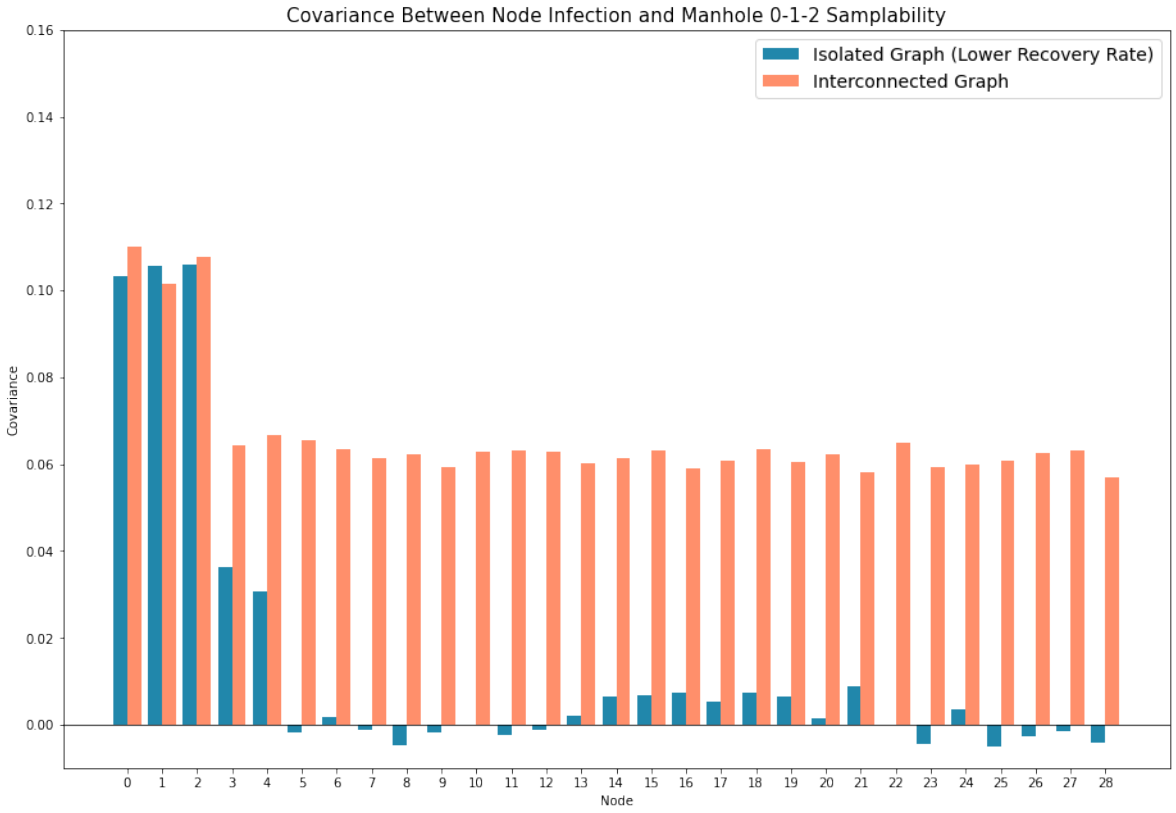

Fig. 7: Comparison of covariances for the isolated and interconnected models

As can be observed, in the isolated graph, only a few nodes display a high covariance with the manhole containing a positive sample, while the other nodes are basically independent. Meanwhile, in the interconnected graph, all the nodes display a positive covariance with the manhole. The fact that the base infection rate has been made constant for both simulations implies that this change is purely a direct result of the extra connections, and not the result of a higher infection rate.

This has implications about our ability to gain information from manhole sampling. In an isolated scenario, obtaining a positive sample from a manhole implies that only a few individuals are likely to be infected and need to quarantine. In contrast, in an interconnected scenario, obtaining a positive sample implies that almost anyone could be infected and needs to quarantine. Wastewater surveillance therefore seems like a more effective means of monitoring disease spread if individuals are isolating.

These results were consistent across manholes, with some limitations. Looking at manhole 14-15-16-17, for instance, gave the results shown in Figure 8.

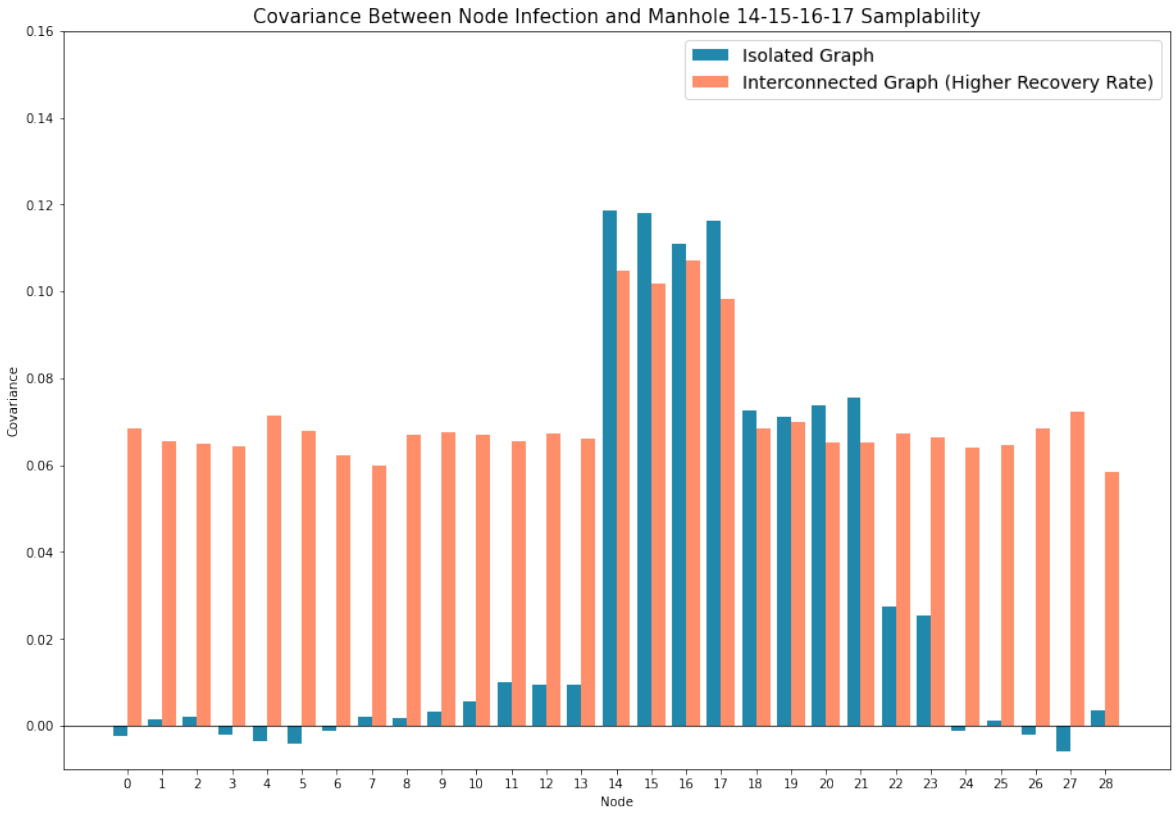

Fig. 8: Comparison of covariances with manhole 14-15-16-17

Note that sewer manholes are named such that they contain all the SIR nodes that feed into them. Here, we can see nodes 14, 15, 16, and 17 have the highest covariance with the manhole in both graphs, followed by 18, 19, 20, and 21, which are downstream. After that, the covariances in the isolated model are either weakly positive or essentially independent, while the covariances in the interconnected model all still display positive correlation. In other words, the same behavior exhibited by node 0-1-2 is also exhibited by 14-15-16-17.

Not all nodes exhibit this behavior, however. Nodes 0-1-2 and 14-15-16-17 are similar in that they are both in the branches at the top of the sewer line where only nodes immediately around them can contribute a positive sample. Other nodes, such as node 4-6-7, are located at junctions in the sewer line. This means that nodes 4, 6, and 7 contribute to this manhole, of course, but also that nodes from two separate "branches" of the graph contribute to it as well. The result is that it is harder to gain information from this node, as shown below in Figure 9.

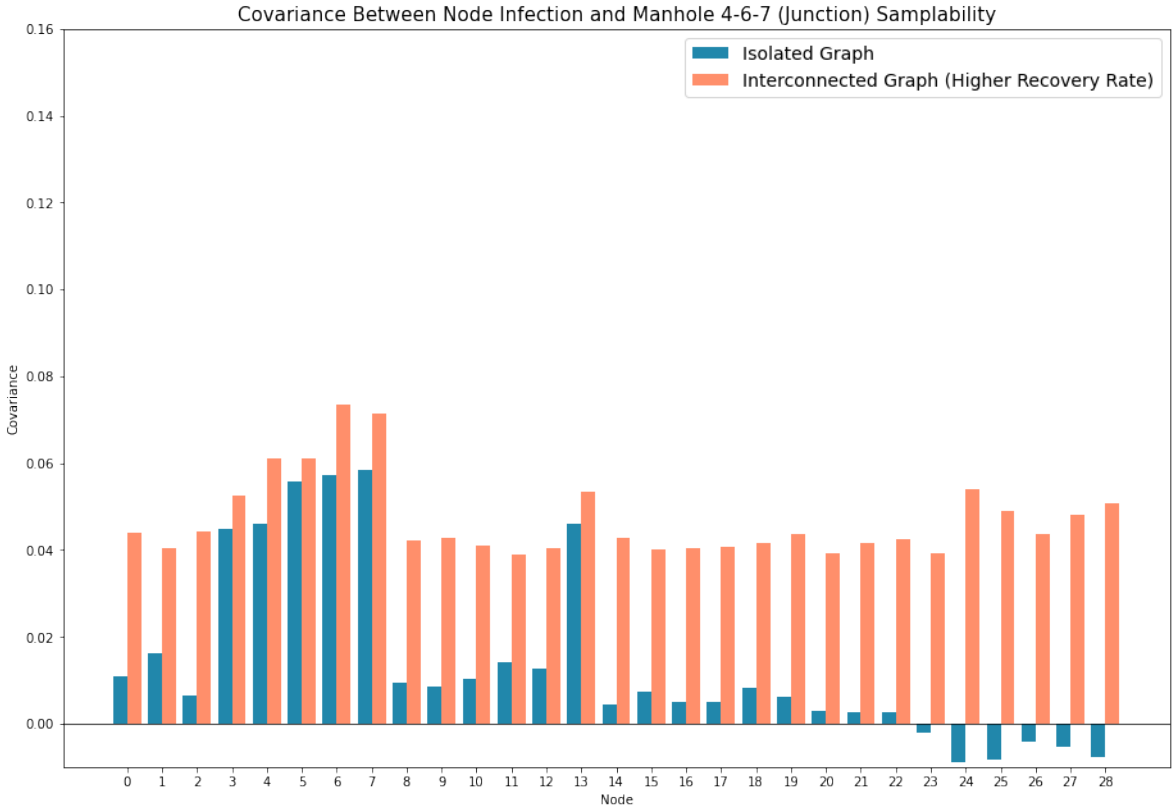

Fig. 9: Comparison of covariances with manhole 4-6-7

This time, the covariances are much lower in both scenarios and much more spread out in the isolated graph. We see high covariances with nodes 4, 6, and 7 as expected, but also high covariances with nodes 3, 5, and 13 as well. This is likely the result of the fact that, while these nodes do not directly feed into node 4-6-7, they do feed into manhole nodes that are one step removed from 4-6-7.

Even so, we can also see that the same general trend from before still holds true, with the isolated graph displaying a positive covariance with only a few nodes, while the interconnected graph displays a positive covariance across before. It is fair to say, at this point, that a higher interconnectivity in the graph leads to less effective wastewater surveillance.

7 Conclusion

Across the board, we observed that collecting a positive sample from a manhole allowed us to identify some nodes as being more likely to be infected in the isolated scenario, while collecting a positive sample in the interconnected scenario implied that almost anyone could

be infected. Translated into the real world, these results show that wastewater monitoring is more effective the less interaction there is between individuals.

As with any conclusion, however, there are limitations. Firstly, it must be acknowledged that these results cannot stand for themselves, as they were created from the result of a model with specific assumptions. Specifically, it could be the case that the assumption that an SIR model would yield accurate results was incorrect. Perhaps an SEIR model, with infected nodes that self-isolate even in the interconnected scenario, would show that wastewater monitoring can still be effective even if communities are very intertwined. Further research could therefore be conducted in this area.

Another important limitation to make mention of is that this investigation only scratched the surface of what can be accomplished with a simulation like this. We only made use of two graphs to conduct this analysis, but theoretically one could make a myriad of graphs each with different properties to continue investigating which ones yield the best results.

Perhaps the best example of a new investigation that could be started with a model like this came at the very end. We found that obtaining a sample from junction 4-6-7 was dubious as a means of determining if any specific nodes were infected. However, this only means that the most straightforward way of wastewater monitoring is ineffective at junctions, and it could be that a more complicated technique would work better. Figure 10 below shows what the probabilities look like if we measure both if 4-6-7 contains a positive sample and if the two sewer nodes that feed into it test negative.

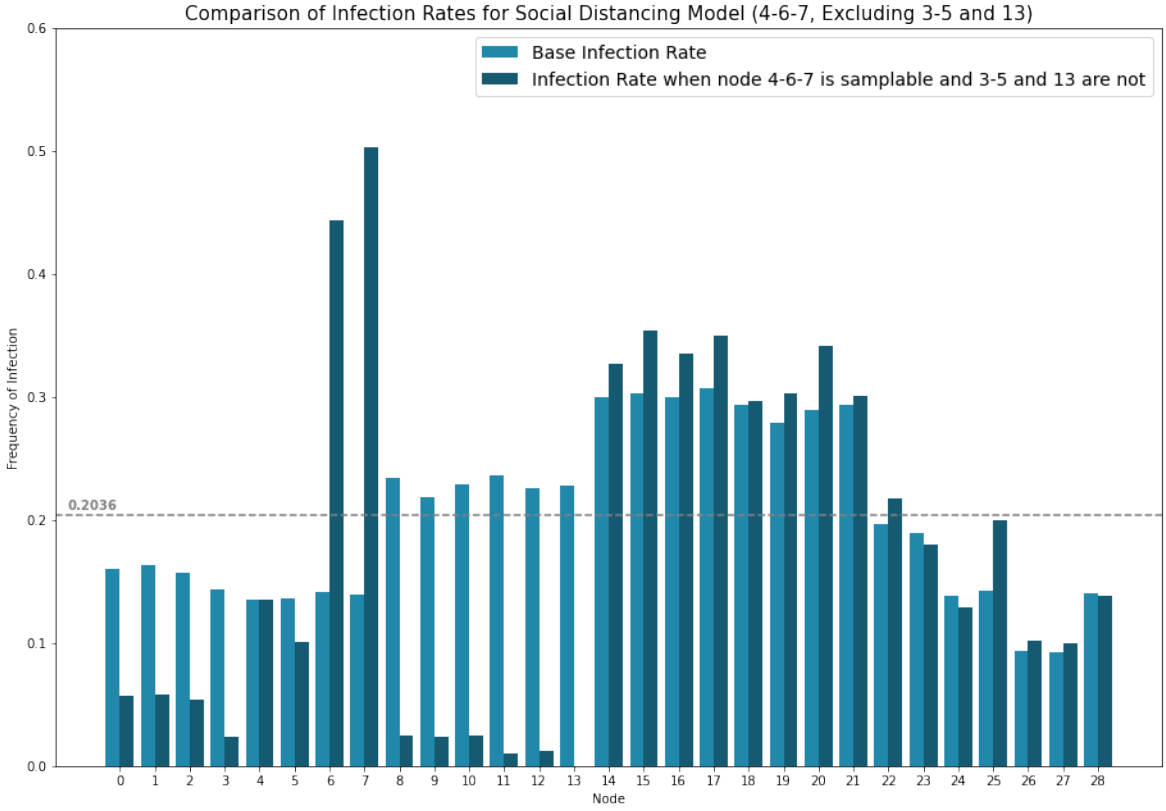

Fig. 10: Probabilities of infection when testing multiple manholes at once

Testing multiple manholes at once gives very interesting results. Nodes 6 and 7 display the type of behavior one might expect, with a high probability when the sample is detected, while nodes 8, 9, 10, 11, 12, and 13 have incredibly low probabilities. This makes sense, as, if node 13 is not samplable, that is a good indicator that nobody in that community is infected, as their samples would have flowed down into that manhole. The strange part of this graph is node 4, who also feeds into manhole 4-6-7, but whose probability did not change. It is possible that this is a result of the graph, with nodes 6 and 7 having high edge weights in their connections to node 5, while node 4 is not a part of their system. This paper did not investigate deeper than that into this problem, but, clearly, it has potential for a future investigation as well.

Even still, the results of this investigation give insight into the nature of wastewater monitoring. Although there is still room for investigation, evidence was found that wastewater surveillance can be made more effective through more social isolation.

References

[1] Wade, M. J., Lo Jacomo, A., Armenise, E., Brown, M. R., Bunce, J. T., et al. *Understanding and managing uncertainty and variability for wastewater monitoring beyond the pandemic: Lessons learned from the United Kingdom national COVID-19 surveillance programmes.* Journal of Hazardous Materials, 424. 2022 February 15.

[2] Biocomplexity Institute, Homepage. University of Virginia. 2022

[3] Li GZ, Haddadan A, Li A, Marathe M, Srinivasan A, Vullikanti A, Zhao Z. *Theoretical Models and Preliminary Results for Contact Tracing and Isolation.* Proceedings of the 21st International Conference on Autonomous Agents and Multiagent Systems. 2022 May 9; 1672-1674.

[4] Adiga, A., Dubhashi, D., Lewis, B. et al. *Mathematical Models for COVID-19 Pandemic: A Comparative Analysis.* Journal of the Indian Institute of Science, Volume 100. 30 October 2020; 793–807.

[5] Solano, Fernando Krause, Steffen, and Wöllgens, Christoph. *An Internet-of-Things Enabled Smart System for Wastewater Monitoring*. IEEE Access, Volume 10. 5 January 2022; 4666-4685

[6] Wei-An Chen, Jongyeon Lim, Shohei Miyata, and Yasunori Akashi. *Methodology of evaluating the sewage heat utilization potential by modelling the urban sewage state prediction model*. Sustainable Cities and Society, Volume 80. 2022

A Appendix: GitHub Repository

The repository can be found here, or at the following URL:

https://github.com/colincrowe/capstone

The source code for the simulation can be found in the file titled `capstone_utils.py`, while the code to generate the results can be found in the two jupyter notebooks.

The input parameters for the isolated graph are found in `fork3.txt`, and the parameters for the interconnected graph are found in `fork3_intercon.txt`. Note that these files are parsed a specific way by the implementation; see the `readme` at the repository for an explanation.