

Soluções para problemas reais envolvendo otimização: estudo sobre topologias conexas para controle de doença em um grafo de infecção

Pedro G. Caninas^{1,2}, Álvaro J.P. Franco¹, Priscila C. Calegari¹

¹Departamento de Informática e Estatística

Universidade Federal de Santa Catarina (UFSC) – Florianópolis - SC - Brasil

²Bolsista do CNPq - Brasil

pedro.caninas@grad.ufsc.br, alvaro.junio, priscila.calegari@ufsc.br

Resumo. Neste trabalho utilizamos um modelo epidemiológico sobre grafos. As pessoas são representadas e classificadas de acordo com os grupos definidos pelo modelo epidemiológico, baseado no modelo (SIR, Suscetíveis-Infectados-Recuperados). O modelo considera uma certa movimentação de pessoas e uma medida de distanciamento social. Chamamos um grafo com todas essas características de grafo de infecção. Realizamos experimentos que ilustram a dinâmica do espalhamento de uma doença sobre um grafo de infecção. Os experimentos consideram dois grafos de infecção que capturam a topologia de duas cidades do Brasil, considerando a vizinhança entre bairros. Os experimentos também consideram árvores geradoras desses grafos de infecção buscando entender o espalhamento de uma doença nesses subgrafos minimamente conexos. Os resultados sugerem estratégias para a redução do número de infectados em uma epidemia.

Palavras-chave: grafo de infecção; árvore geradora; modelo epidemiológico; teoria dos grafos; modelo SIR.

1. INTRODUÇÃO

Recentemente presenciamos o surgimento da pandemia de Covid-19, uma onda de infecção pelo coronavírus SARS-CoV-2 que atingiu o mundo inteiro. Países rapidamente tomaram medidas de contenção para controlar seu espalhamento. Nesse contexto, a importância de se prever a evolução de uma pandemia veio à tona, já que esta previsão nos permitiria tomar decisões de forma inteligente. Modelos epidemiológicos são ferramentas amplamente utilizadas na epidemiologia para estudar e prever o comportamento de uma doença em uma população. Existem diferentes tipos desses modelos, que se encarregam de separar as pessoas em grupos de acordo com a sua condição relativa à doença e definir quantitativamente a dinâmica entre estes grupos, como os modelos clássicos SIS (*Suscetíveis-Infectados-Suscetíveis*) e SIR, mais detalhes em [Hethcote 1989]. Esses modelos podem ser aplicados em uma população de forma global, mas é possível adaptar tais modelos para serem utilizadas sobre grafos, uma estrutura composta de vértices e arestas, onde a interação entre as pessoas pode ser modelada localmente. Um grafo pode modelar relações entre objetos, sendo possível realizar diversas mudanças em sua topologia, reproduzindo medidas de contenção reais, como o isolamento social.

Por causa da pandemia de Covid-19, diversos modelos epidemiológicos foram utilizados para estudar e prever a dinâmica de doenças. O trabalho de [Barbosa et al. 2021] faz uma análise da pandemia na cidade de Marabá utilizando o modelo SEIR (*Suscetíveis-Expostos-Infectados-Recuperados*), com dados reais da cidade e da doença. Outros trabalhos, como o [Manrique-Abril et al. 2020], consideram uma abordagem SIR puramente matemática para analisar e prever o desenvolvimento da Covid-19 no território colombiano. O trabalho [Visbal and Pedraza 2020] também utiliza o modelo SIR e realiza uma conexão entre os parâmetros do modelo e as motivações em torno das medidas de contenção adotadas na cidade de Santa Marta, Colômbia. Os modelos epidemiológicos também podem ser aplicados sobre grafos, como feito no trabalho [Keeling and Eames 2005]. Nele, cada pessoa é representada como um vértice da rede e a proximidade/contato entre pessoas são representadas como arestas, individualizando de fato as pessoas.

Neste trabalho utilizaremos um modelo epidemiológico sobre um grafo com restrições na circulação de pessoas, desenvolvido em [Franco 2020] tendo como base um modelo epidemiológico sobre um autômato celular [White et al. 2007]. Nele vértices contêm uma certa quantidade de pessoas, representando um bairro. O modelo epidemiológico considerado foi o SIR, em que as pessoas são classificadas como *suscetíveis*, *infectadas* e *recuperadas*, sendo aplicado em cada vértice do grafo. As pessoas em um bairro podem se movimentar entre bairros adjacentes da sua origem e cada pessoa de um vértice está no correspondente grupo SIR. Além disso, as pessoas podem aderir ou não a um *distanciamento social*, uma medida dada por um índice de distanciamento por vértice.

Em [Franco 2020], foram desenvolvidos experimentos sobre grafos com diferentes topologias. Alguns exemplos são a topologia *completa* (todos os vértices do grafo estão conectados entre si), a topologia *grade* (o grafo conecta os vértices em forma de uma matriz bidimensional) e a topologia *linha* que consiste em um grafo que é um caminho simples. O autor constatou experimentalmente que o comportamento de uma doença é diferente dependendo da topologia do grafo, com as mesmas condições iniciais e restrições de circulação de pessoas. Os grafos gerados tinham 100 vértices com 3.000 pessoas cada e inicialmente haviam 400 pessoas infectadas em um deles.

No presente trabalho, realizamos os experimentos sobre dois grafos que representam regiões adjacentes de duas cidades do Brasil - Rio de Janeiro e Florianópolis. Nosso objetivo aqui é analisar a dinâmica de uma doença sobre esses grafos, após realizar alterações na topologia do grafo. Os números de pessoas em cada vértice desses grafos (população de um bairro) foram obtidos através de dados oficiais (IBGE e Data.Rio). Os experimentos realizados compararam o desenvolvimento de uma doença ao longo do tempo sobre os grafos originais com diferentes subgrafos conexos. Na grande maioria dos casos, houve maior controle sobre o espalhamento de uma doença sobre um grafo quando a topologia dele é alterada para uma topologia minimamente conexa. Por último realizamos uma breve análise de convergência sobre o modelo, observando as consequências de sua discretização em intervalos de tempo menores.

2. MATERIAL E MÉTODOS

Nesta seção, apresentamos o funcionamento do modelo. Em seguida detalhamos a construção das instâncias (*grafos de infecção*), a obtenção dos dados e a sua

implementação. Além disso, explicamos brevemente alguns dos algoritmos utilizados nos experimentos.

2.1. Apresentação do modelo

Para entender o modelo é necessário primeiro conhecer um pouco mais sobre a estrutura de um grafo. Um grafo é composto por vértices e arestas, representando um objeto e um certo relacionamento com outros objetos, respectivamente. A Figura 1 ilustra um exemplo de um grafo onde vértices são bairros e a existência de uma fronteira entre eles define uma aresta.

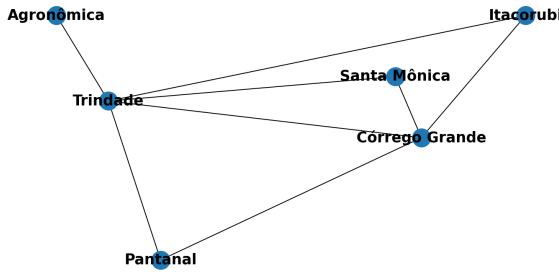


Figura 1. Grafo com alguns bairros próximos da Universidade Federal de Santa Catarina (UFSC)

Utilizamos a notação de conjuntos para descrever o modelo. O grupo SIR de pessoas no vértice i e no tempo t (em dias) é dado por $N_i^t = S_i^t \cup I_i^t \cup R_i^t$. O subconjunto $\dot{S}_i^t \subseteq S_i^t$ ($\ddot{S}_i^t \subseteq S_i^t$) destaca as pessoas suscetíveis do vértice i no tempo t que atendem (não atendem) ao distanciamento social. Da mesma forma temos isso para os infectados \dot{I}_i^t (\ddot{I}_i^t) e recuperados \dot{R}_i^t (\ddot{R}_i^t). A constante α_i^t corresponde ao índice de pessoas do vértice i que atendem ao distanciamento social no tempo t . Com isso, o número de suscetíveis que atendem e não atendem ao distanciamento é, respectivamente $|\dot{S}_i^t| = |\alpha_i^t|S_i^t|$ e $|\ddot{S}_i^t| = |S_i^t| - |\dot{S}_i^t|$. O mesmo é aplicado para obter os números de infectados e de recuperados que atendem e não atendem ao distanciamento social. Portanto, as pessoas de i e no tempo t que atendem ao distanciamento social pertencem ao conjunto $\dot{N}_i^t = \dot{S}_i^t \cup \dot{I}_i^t \cup \dot{R}_i^t$; e as pessoas que não atendem ao distanciamento social pertencem ao conjunto $\ddot{N}_i^t = \ddot{S}_i^t \cup \ddot{I}_i^t \cup \ddot{R}_i^t$.

Neste modelo, a movimentação das pessoas acontece da seguinte forma. Existem pessoas circulando em cada vértice i do grafo e em cada instante de tempo t . Essas pessoas são pessoas do próprio vértice i (de \dot{N}_i^t e de \ddot{N}_i^t) e pessoas de vértices adjacentes de i que não atendem ao distanciamento social (de $\bigcup_{j \in \chi(i)} \ddot{N}_j^t$, sendo $\chi(i)$ o conjunto dos vértices adjacentes ao vértice i). Um parâmetro $\beta_{j \rightarrow i}^t$ é utilizado para representar a proporção de pessoas do vértice i que vieram do vértice j , sendo $j \in \chi(i)$. Cada aresta $\{i, j\}$ possui dois parâmetros distintos, $\beta_{j \rightarrow i}^t$ e $\beta_{i \rightarrow j}^t$, ambos no intervalo $[0, 1]$.

O conjunto de pessoas suscetíveis que encontram com uma pessoa infectada no vértice i e no instante de tempo t é denotado por X_i^t . Considerando o distanciamento social, temos \dot{X}_i^t (\ddot{X}_i^t) o conjunto de pessoas $p \in \dot{S}_i^t$ ($p \in \ddot{S}_i^t$) que encontram com um infectado no vértice i (em vértices adjacentes a i). Note que o modelo considera uma movimentação de pessoas que atendem ao distanciamento social. Tais pessoas precisaram

se movimentar por algum motivo. Dessa forma, temos $X_i^t = \dot{X}_i^t \cup \ddot{X}_i^t$. Finalmente, o modelo sobre o grafo considerando o modelo epidemiológico SIR, conforme [Franco 2020], em cada vértice é dado por,

$$\begin{aligned} |I_i^{t+1}| &\geq |I_i^t| - \lceil \varepsilon \mathbb{E}[|I_i^t|] \rceil + \lfloor v(\mathbb{E}[|\dot{X}_i^t|] + \mathbb{E}[|\ddot{X}_i^t|]) \rfloor =: |I^{t+1}|, \\ |S_i^{t+1}| &\leq |S_i^t| - \lfloor v(\mathbb{E}[|\dot{X}_i^t|] + \mathbb{E}[|\ddot{X}_i^t|]) \rfloor =: |S_i^{t+1}|, \\ |R_i^{t+1}| &\leq |R_i^t| + \lceil \varepsilon \mathbb{E}[|I_i^t|] \rceil =: |R_i^{t+1}|, \end{aligned} \quad (1)$$

sendo v a taxa de virulência e ε a taxa de recuperação. O valor esperado de $|\dot{X}_i^t|$ é definido como $\mathbb{E}[|\dot{X}_i^t|] = \dot{S}_i^t \mathbb{P}\{\dot{Y}_{pi}^t\}$, com \dot{S}_i^t sendo o número de pessoas suscetíveis de i que atendem ao distanciamento social mas que por algum motivo circulam somente em i . O valor esperado de $|\ddot{X}_i^t|$ é definido como $\mathbb{E}[|\ddot{X}_i^t|] = \ddot{S}_i^t \mathbb{P}\{\ddot{Y}_{pi \rightarrow i}^t\} + \sum_{j \in \chi(i)} \ddot{S}_j^t \mathbb{P}\{\ddot{Y}_{pi \rightarrow j}^t\}$, com \ddot{S}_i^t o número de pessoas suscetíveis de i que não atendem ao distanciamento social e que circulam em i e \ddot{S}_j^t , o número de pessoas suscetíveis de i que não atendem ao distanciamento social e que circulam em um vértice adjacente $j \in \chi(i)$. As probabilidades $\mathbb{P}\{\dot{Y}_{pi}^t\}$, $\mathbb{P}\{\ddot{Y}_{pi \rightarrow i}^t\}$ e $\mathbb{P}\{\ddot{Y}_{pi \rightarrow j}^t\}$ são todas iguais a razão entre o número de pessoas infectadas circulando em um vértice e o número total de pessoas circulando em tal vértice. Além disso, $\lceil \cdot \rceil$ e $\lfloor \cdot \rfloor$, representam respectivamente as funções teto e piso. Dessa forma, temos para cada instante de tempo e para cada vértice do grafo, limitantes inferiores para o número de infectados e limitantes superiores para os números de suscetíveis e recuperados. Para a execução dos experimentos, os números de infectados, suscetíveis e recuperados são fixados nos limitantes determinados.

2.2. Estruturação do Grafo e obtenção dos dados

A primeira etapa na construção do grafo de infecção é conhecer qual a sua estrutura. De acordo com [Franco 2020] temos bairros sendo representados por vértices e arestas representando existência de uma fronteira entre dois bairros. Uma fronteira, neste caso, é definida como a possibilidade de uma pessoa transitar diretamente de um bairro a outro.

A instância de Florianópolis utilizada foi a mesma de [Ataide 2022], concedida pelo autor para nossos experimentos. O trabalho detalha a construção do grafo e a obtenção dos dados necessários. Nele os bairros da cidade foram agrupados, ou seja, vértices correspondem a grupos de bairros vizinhos. São 16 grupos de bairros e 25 arestas interligando os grupos. Os grupos combinam bairros que estão em regiões próximas uns dos outros e foram assim definidos: Grupo 1: Cachoeira do Bom Jesus, Ingleses, Canasvieiras, Pontas das Canas, Santinho; Grupo 2: Vargem Grande, Rio Vermelho; Grupo 3: Vargem Pequena, Jurerê; Grupo 4: Santo Antônio de Lisboa, Ratones, Saco Grande, Costa da Lagoa; Grupo 5: Itacorubi, João Paulo, Lagoa da Conceição, Barra da Lagoa; Grupo 6: Balneário, Jardim Atlântico, Estreito, Novo Continente; Grupo 7: Coloninha, Sapé, Monte Cristo, Capoeiras; Grupo 8: Vila Aparecida, Abraão, Coqueiros; Grupo 9: Agronômica, Praia do Centro, Monte Serrat; Grupo 10: Trindade, Pantanal, Saco dos Limões; Grupo 11: Rio Tavares, Costeira do Pirajubaé; Grupo 12: Canto da Lagoa, Córrego Grande; Grupo 13: Tapera, Carianos, Fazenda do Rio Tavares, Campeche; Grupo 14: Morro das Pedras, Armação; Grupo 15: Ribeirão da Ilha, Alto Ribeirão; Grupo 16: Caieira da Barra do Sul, Pântano do Sul. A Figura 2 apresenta o grafo simplificado da cidade de Florianópolis, com base na divisão de grupos descrita anteriormente.

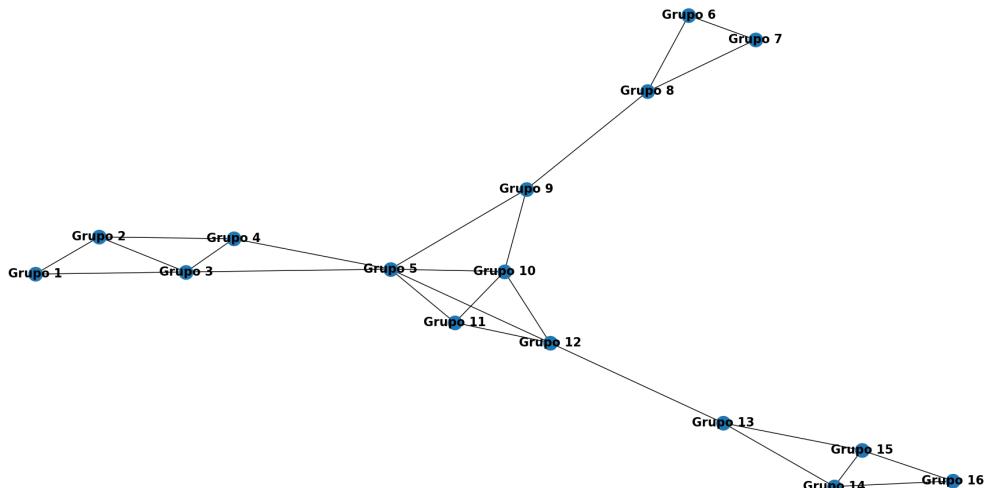


Figura 2. Grafo simplificado de Florianópolis

No presente trabalho, apresentamos a instância da cidade do Rio de Janeiro. Os dados foram coletados sobre os bairros e suas adjacências através do mapa do plano diretor da cidade, apresentado na Figure 3. A partir destes dados criamos a lista de adjacências, um arquivo que lista todos os bairros e seus vizinhos. Com estes dados já é possível visualizar o grafo, conforme apresentado na Figura 4.

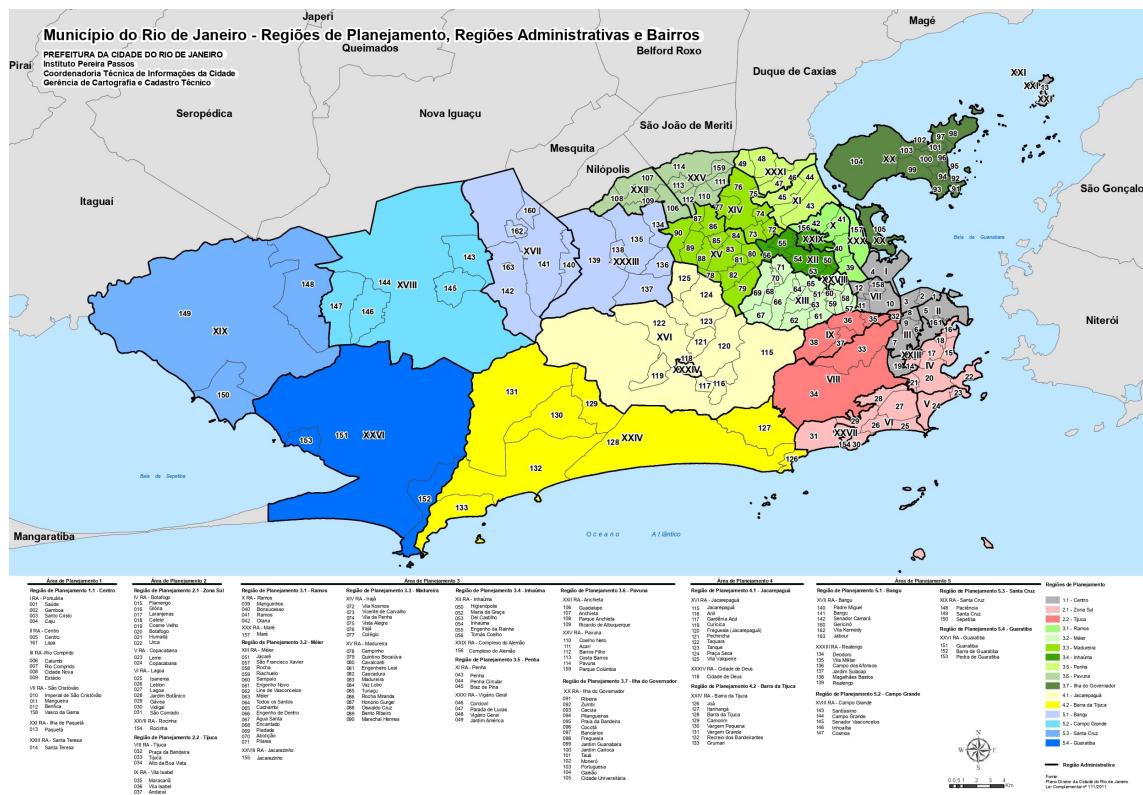


Figura 3. Município do Rio de Janeiro
Fonte: Plano Diretor da Cidade do Rio de Janeiro Lei complementar nº 111/2011

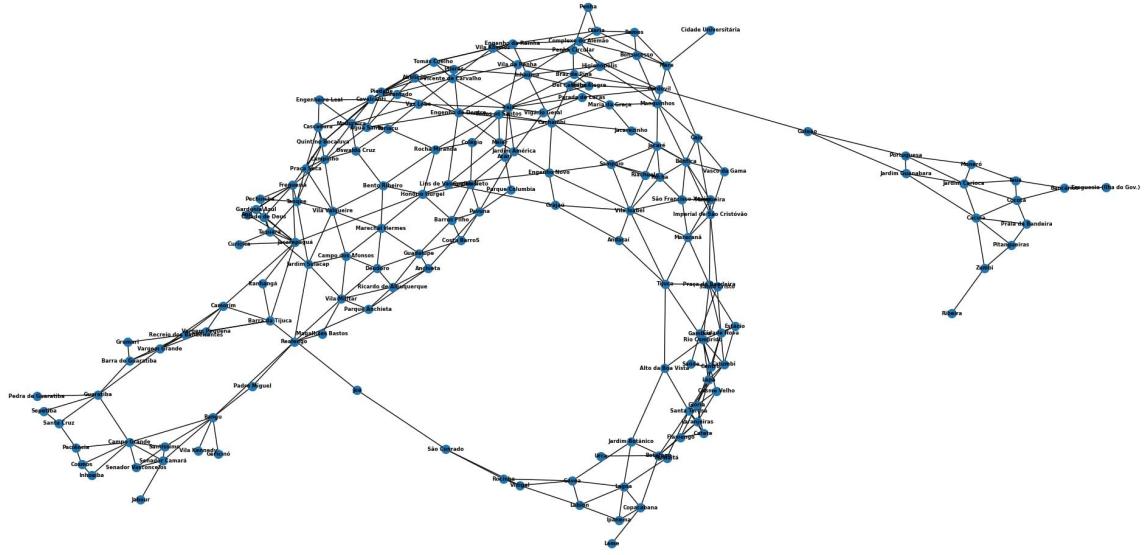


Figura 4. Grafo do Rio de Janeiro

Antes da implementação do modelo, foi necessário obter os dados da população de cada bairro da cidade do Rio de Janeiro. O Instituto Municipal de Urbanismo Pereira Passos, referência na divulgação de dados geográficos para o planejamento urbano e produção cartográfica do Rio, mantém o portal Data.Rio. Neste portal, são disponibilizados diversos dados da cidade, incluindo tabelas com os bairros do Município do Rio e suas respectivas populações. Os dados do portal foram extraídas a partir dos dados do Censo Demográfico do Brasil de 2010 realizado pelo Instituto Brasileiro de Geografia e Estatística (IBGE) [Rio 2010b], [Rio 2010a], [Rio 2010c]. Por terem sido coletados em 2010, os dados não incluem a população dos bairros Lapa, Vila Kennedy e Jabour, já que eles não tinham sido criados. Portanto a lista de adjacências teve que ser alterada para entrar em conformidade com a tabela de população. Além deles, o bairro Paquetá também foi retirado, já que é uma ilha sem acesso por ruas, apenas através de barcas, com dados escassos do fluxo de pessoas, o que tornaria a modelagem da movimentação mais complicada. Com as tabelas em mãos e feitos os ajustes necessários nas listas de adjacências, foi criado o arquivo *grafo*, onde cada linha representa um bairro, com seu nome seguido de seus bairros vizinhos (lista de vértices adjacentes) e propriedades, por exemplo:

Copacabana, Ipanema, Botafogo, Leme, Lagoa/24, 146392, 0.2

As propriedades (após a barra) são, respectivamente, o ID do bairro, sua população e o parâmetro *beta*, que representa a proporção de pessoas de circulam em um bairro e que vieram de bairros vizinhos. Os nomes dos bairros foram retirados do mapa da Figura 3, a população foi obtida pela leitura da tabela 2973 do site Data.Rio [Rio 2010b] e o parâmetro β calculado a partir da expressão $\beta_{j \rightarrow i}^t = \frac{1}{|\chi(i)|+1}$, como descrito em [Franco 2020], sendo $|\chi(i)|$ o número de bairros vizinhos (vértices adjacentes) do bairro (vértice) *i*. A partir desse arquivo, o grafo pode ser facilmente criado e modificado, mudando o seu *script* de criação, atualizando os dados que ele utiliza (como as populações) ou até mesmo trocando a cidade a ser utilizada.

2.3. Implementação do Modelo

A implementação foi feita em *Python*, utilizando a biblioteca *Networkx* para a criação e manipulação do grafo. Essa biblioteca permite criar um grafo, adicionar propriedades a vértices e arestas, além de facilitar a sua apresentação gráfica.

Todo o modelo é implementado em uma classe que necessita apenas de um argumento, o caminho (*path*) para o arquivo grafo. Ela possui atributos do modelo, como a taxa de virulência v , a taxa de recuperação ϵ , o fator de distanciamento α e λ (fração de pessoas que respeitam o distanciamento mas precisam sair e portanto podem ser infectadas \dot{S}_i^t) com valores diferentes para cada grupo S, I e R. A classe também possui métodos para construir o grafo e reiniciar seus valores, caso seja necessário. Bem como, funções para gerar graficamente o grafo e os gráficos apresentados neste trabalho. A função de avançar tempo é responsável por avançar t (medido em dias, $t \in \mathbb{N}$), simulando o modelo de fato. Nela, em cada um dos dias, três ações são realizadas. Primeiramente movimenta-se as pessoas entre os vértices, como descrito na Seção 2.1. Depois o processo de infecção começa, sendo calculada a probabilidade $\mathbb{P}\{Y_{pi}^t\}$ para determinar os valores de encontros daquele dia, ou seja, \dot{X}, \ddot{X} . Na segunda ação, as quantidades de cada subgrupo de distanciamento ($\dot{S}I\dot{R}, \dot{S}\ddot{I}\dot{R}$ e $\ddot{S}\ddot{I}\dot{R}$) é atualizada por meio do sistema de equações (1), apresentado na Seção 2.1). A terceira ação, no final do dia, é regressar as pessoas de volta aos seus vértices de origem, terminando o dia e passando para o próximo dia t . Essa movimentação gera o espalhamento da infecção ao longo do grafo, onde o infectado de um vértice que foi para outro, pode infectar um suscetível desse vértice vizinho (ou suscetível do próprio vértice que saiu para esse vizinho). No dia seguinte, este novo infectado poderá (ou não) visitar os seus vértices vizinhos, criando o efeito cascata.

A Figura 5 apresenta a composição de um vértice após a primeira movimentação. Os grupos de pessoas Suscetíveis, Infectados e Recuperados, são subdivididos em subgrupos de distanciamento que participam do espalhamento da infecção: pessoas que respeitam o distanciamento social, mas precisam sair de casa ($\dot{S}, \dot{I}, \dot{R}$), pessoas que não respeitam o distanciamento social e não saíram do próprio vértice ($\ddot{S}, \ddot{I}, \ddot{R}$) e pessoas que não respeitam o distanciamento social que vieram de outros vértices ($\ddot{S}, \ddot{I}, \ddot{R}$). A seção do gráfico em destaque mostra de onde vieram tais suscetíveis. Cada vértice armazena a origem de uma pessoa para, ao final do dia, devolvê-la ao bairro de origem correto.

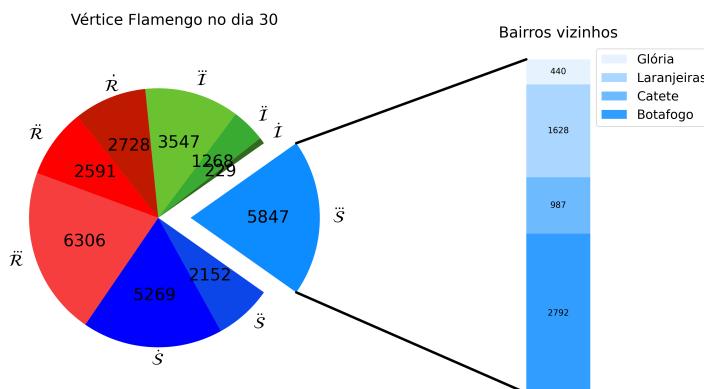


Figura 5. Composição do vértice Flamengo em $t = 40$

Durante a simulação do modelo são armazenados também os valores totais de cada grupo SIR e seus respectivos instantes de tempos t para a visualização gráfica. A Figura 6 apresenta o gráfico resultante da simulação numérica do modelo no período de 200 dias. Nesta simulação foi escolhido o vértice do bairro do Flamengo com um número de infectados no instante de tempo inicial t_0 e pico no número de infectados ocorre por volta do dia 75 com aproximadamente 900 mil infectados.

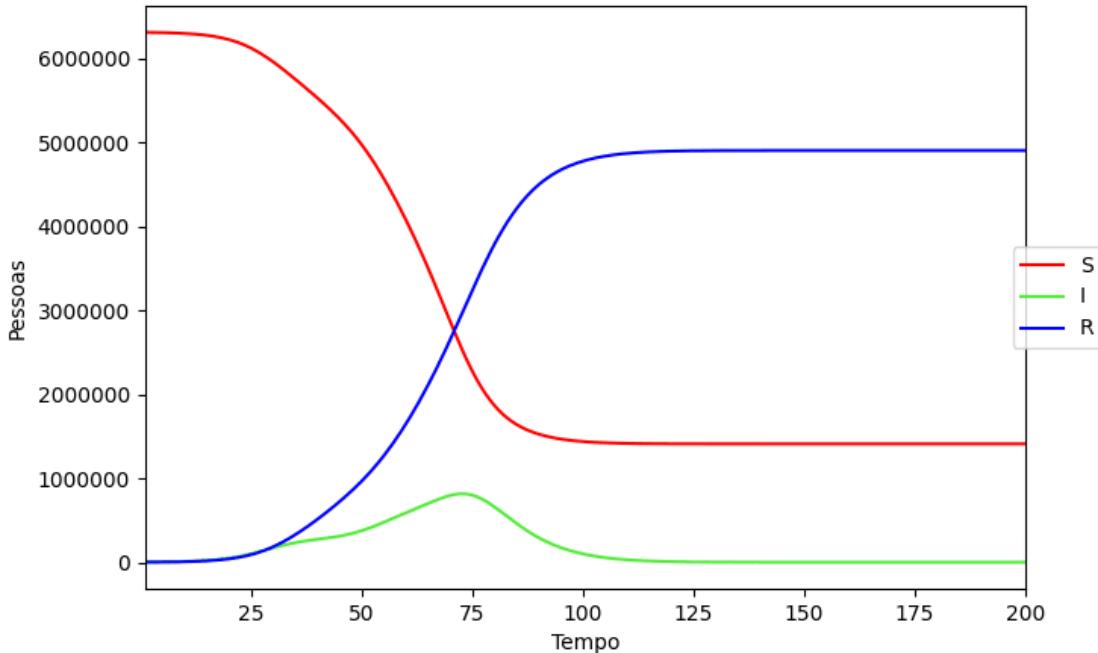


Figura 6. Gráfico de Infecção ao longo de 200 dias

Para **todos** os nossos experimentos numéricos utilizamos os seguintes valores de parâmetros: fração de infectados iniciais $= \frac{1}{20}$, taxa de virulência $v = \frac{91}{200}$, taxa de recuperação $\varepsilon = \frac{29}{200}$, fator de distanciamento $\alpha = \frac{2}{5}$, e fração de pessoas que respeitam o distanciamento, mas precisam se movimentar $\lambda_S = \frac{2}{5}$, $\lambda_I = \frac{1}{10}$, $\lambda_R = \frac{3}{5}$. Para demonstrar o funcionamento do modelo, foram gerados gráficos de pizza para cada bairro com seus valores SIR (Suscetíveis na cor azul, Infectados na cor verde e Recuperados na cor vermelha) em diferentes dias. A Figura 7 apresenta 4 instantes de tempo, os dias 1, 50, 75 e 200 da simulação do experimento, apresentado na Figura 6.

O primeiro gráfico apresenta o estado inicial dos vértices, com suscetíveis em todos os vértices com exceção de um vértice, que contém uma pequena quantidade de infectados (5% da população). Já o segundo gráfico apresenta o dia 50 da evolução e espalhamento da doença, tendo vértices com muitas pessoas recuperadas, alguns com vários infectados e vértices onde a infecção ainda está começando, por exemplo, a antepenúltima linha. O terceiro gráfico mostra o pico da doença, com diversos infectados em vários vértices. No último gráfico podemos observar o final do espalhamento da doença, com nenhum infectado presente. Cerca de 20% de suscetíveis restaram, indicando uma certa imunidade de rebanho, fazendo com que a doença não conseguisse continuar se

espalhando.

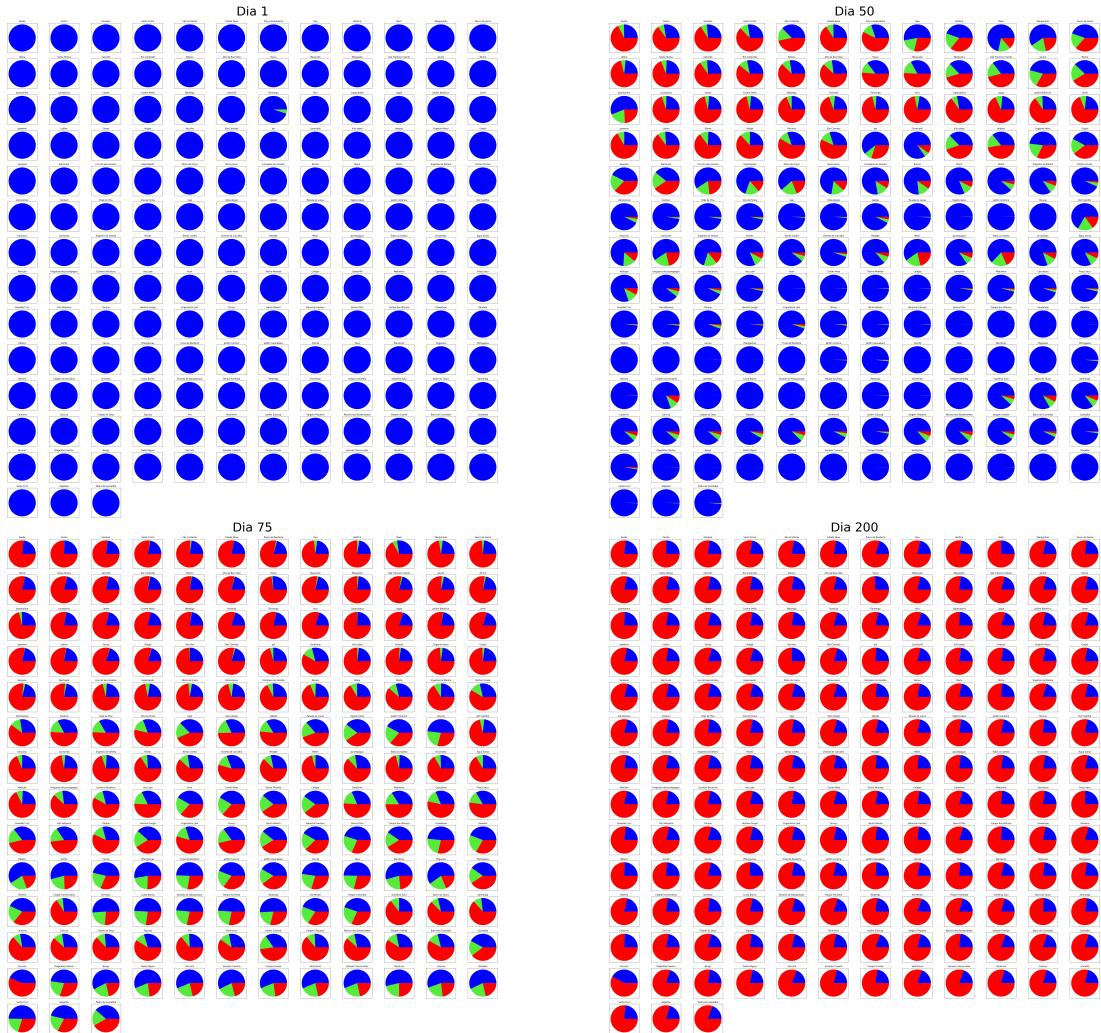


Figura 7. Gráficos pizza do SIR por vértice em diferentes dias no modelo

Os gráficos gerados por esta simulação podem ser encontrados em uma animação disponível aqui¹.

2.4. Problema de interesse

Após os primeiros experimentos, identificamos um instante de tempo em que ocorre o pico no número de infectados, veja a Figura 6. Nossa objetivo é tentar diminuir o máximo que conseguirmos esse pico, removendo as arestas do grafo de infecção, mas sem deixá-lo desconexo. Em um primeiro momento, nós realizamos essas remoções manualmente, seguindo algum critério. Depois, as remoções foram feitas por meio de algoritmos. Os algoritmos de busca em grafos podem ser utilizados para obter subgrafos conexos chamados de árvores geradoras. Tais algoritmos visitam vértices em uma ordem definida. A

¹Link na versão digital deste documento para o endereço:
<https://www.youtube.com/watch?v=vPmUFQH0uig> .

partir disso, podemos criar a árvore ao conectar vértices visitados ao vértice visitado na etapa anterior. A seguir uma breve explicação de como esses algoritmos funcionam, mais detalhes podem ser encontrados em [Feofiloff et al. 2011] e [Dasgupta et al. 2008].

Uma Busca em Profundidade é como visitar um labirinto, você vai andando por um caminho até ficar sem saída. A partir daí, começa a voltar por esse caminho, entrando em todos os outros caminhos que você encontrar nessa volta. Ao entrar em um caminho encontrado na volta, você segue até o fim dele até não encontrar a saída. E assim, executa os mesmos procedimentos anteriores nesse outro caminho. Você não deve visitar nenhum caminho novamente e deve visitá-lo até não poder mais voltar, por já ter visitado todos. Essa lógica é apresentada no Algoritmo 1 escrito em pseudocódigo.

Algoritmo 1: Busca em Profundidade

Explorar(Grafo $G(V, E)$, $v \in V$):

visitado(v) \leftarrow true

para cada vizinho u de v faça

se $\text{não visitado}(u)$ **então**

Explorar(u)

A Figura 8 apresenta um grafo qualquer (à esquerda) e sua árvore de busca em profundidade (à direita) com vértice inicial 1, obtida por meio do Algoritmo 1.

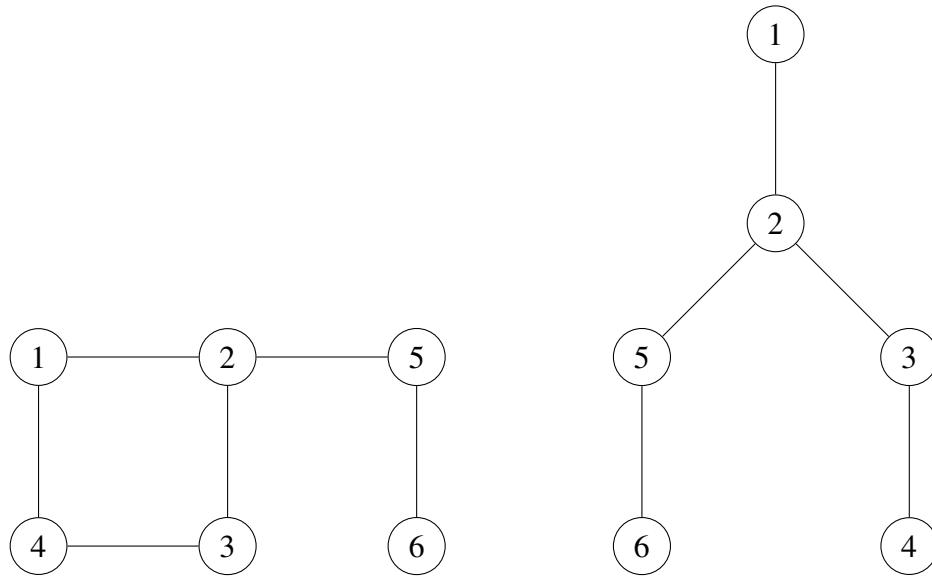


Figura 8. Grafo (à esquerda) e sua árvore de busca em profundidade (à direita).

Ao realizar a busca passo a passo, começamos colocando o vértice inicial, o vértice 1, na árvore e em seguida visitando o seu primeiro vizinho, o vértice 2, e colocando-o na árvore, ligado ao seu antecessor (1). No vértice 2 há dois possíveis caminhos, o algoritmo escolhe o primeiro que encontrar, no caso o caminho que leva ao vértice 3, pelos vizinhos estarem ordenados, sendo este colocado na árvore e ligado ao seu an-

tecessor (2). O vértice 3 só tem um vizinho não visitado, então 4 será visitado, colocado na árvore e ligado ao seu antecessor. Agora não há mais caminhos para se visitar, já que os vértices 1 e 3 já foram visitados. Assim, começaremos a voltar pelo caminho, até que exista um caminho possível de se caminhar. Neste caso apenas quando chegarmos no vértice 2 novamente, teremos um novo vértice não visitado, o vértice 5. Ele será colocado na árvore, ligado ao seu antecessor. Por fim, o vértice 6 será visitado, já que é o único vizinho não visitado de 5, colocado na árvore e ligado ao vértice 4. Como não há mais para onde ir, o algoritmo vai voltando novamente tentando encontrar outro caminho. Ele chegará no vértice inicial e não terá mais para onde ir ou voltar. Dessa forma, o algoritmo termina, formando a árvore de busca em profundidade final.

A Busca em Largura têm um princípio de funcionamento diferente da Busca em Profundidade. Ela recebe um vértice inicial e começa a visitar outros vértices a partir dele. Essa busca visita eles por níveis, visitando todos os vizinhos do inicial, para em seguida visitar os vizinhos desses vizinhos. O processo continua até não ter mais vizinhos para visitar. O Algoritmo 2 apresenta essa lógica utilizando uma fila, onde quem entra primeiro, sai primeiro.

Algoritmo 2: Busca em Largura

Explorar(Grafo G(V, E), $v \in V$):

```

    fila ← [v]
    visitado(v) ← true

    enquanto existir vértice na fila faça
        v ← fila.pop() // remove e retorna primeiro item da
                        fila

        para cada vizinho u de v faça
            se não visitado(u) então
                visitado(u) ← true
                fila ← fila + u

```

A Figura 9 apresenta um grafo (à esquerda, o mesmo apresentado na Figura 8) e sua árvore de busca em largura (à direita) com vértice inicial 1, obtida por meio do Algoritmo 2. Começamos pelo vértice 1, visitando e colocando-o na fila. Em seguida, entramos na repetição e retiramos o item mais antigo da fila, nesse caso o vértice inicial, e visitamos seus vizinhos (vértices 2 e 4), adicionando-os na fila e ligando ao vértice anterior (1). Na próxima iteração superior, o primeiro vizinho visitado (vértice 2) vai ser retirado da fila e seus vizinhos visitados (vértices 3 e 5) são adicionados na fila e ligados a ele. Depois, o próximo vértice da fila será retirado (vértice 4) e como não tem vizinhos, a iteração superior prossegue, retirando o vértice 3 da fila, que também não tem vizinhos. Assim, o vértice 5 será retirado da fila e seu único vizinho (vértice 6) é visitado e colocado na fila e ligado ao seu vértice anterior (5). Por fim, o último vértice restante (6) será retirado da fila e como não tem vizinhos a iteração irá terminar e o algoritmo terá

completado sua execução.

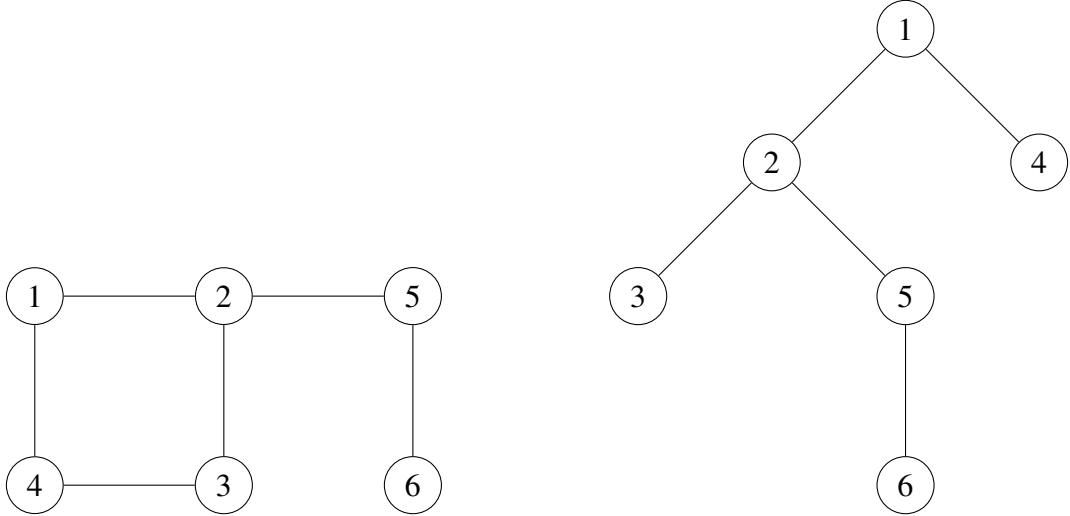


Figura 9. Grafo (à esquerda) e sua árvore de busca em largura (à direita).

Esse algoritmo gera uma árvore com níveis de distância do vértice inicial. No nível 0 temos o vértice 1, no nível 1 os vértices 2 e 4, etc. A distância de 1 até 6, em números de vértices, é exatamente o seu nível na árvore, nesse caso, 3.

2.5. Método de Euler para a discretização do modelo

Para analisar a precisão das aproximações, com intervalos de tempo menores, conforme [Linge and Langtangen 2020], o sistema de equações que rege o nosso modelo epidemiológico (1) é dado por,

$$\begin{aligned} S_i^{k+1} &= S_i^k - \Delta t(v(\dot{X}_i^k + \ddot{X}_i^k)), \\ I_i^{k+1} &= I_i^k + \Delta t(v(\dot{X}_i^k + \ddot{X}_i^k) - \varepsilon I_i^k), \\ R_i^{k+1} &= R_i^k + \Delta t(\varepsilon I_i^k), \end{aligned} \quad (2)$$

sendo $\Delta t = (t_F - t_0)/N$ o tamanho do intervalo de tempo, t_F e t_0 os tempos final e inicial da simulação, k representa o instante de tempo $t = t_0 + k\Delta t$ e N o número de subintervalos no intervalo de tempo $[t_0, t_F]$. Ao dividir o sistema de equações (2) por Δt e fazendo $\Delta t \rightarrow 0$, obtemos um sistema de Equações Diferenciais Ordinárias (EDOs) para cada vértice i do grafo, dado por,

$$\begin{aligned} \frac{dS_i}{dt} &= -(v(\dot{X}_i + \ddot{X}_i)), \\ \frac{dI_i}{dt} &= +\Delta t(v(\dot{X}_i + \ddot{X}_i) - \varepsilon I_i), \\ \frac{dR_i}{dt} &= \Delta t(\varepsilon I_i), \end{aligned} \quad (3)$$

sendo $\frac{dS_i}{dt}$, $\frac{dI_i}{dt}$ e $\frac{dR_i}{dt}$, respectivamente, as derivadas de $S(t)$, $I(t)$ e $R(t)$ em relação ao tempo, ou seja, as taxas de variação de indivíduos suscetíveis, infectados e recuperados

ao longo do tempo. O método de Euler é um método numérico utilizado para aproximar soluções de EDOs, mais detalhes em [Burden and Faires 2011]. Para cada passo Δt , calculamos um valor aproximado em $t + \Delta t$, por meio da reta tangente da solução em t . Se realizarmos passos grandes, o ponto na reta tangente ficará distante do ponto sobre a curva da função solução. Com passos pequenos, a solução aproximada no próximo instante de tempo, ficará mais próxima da solução, diminuindo o erro gerado pelas aproximações sucessivas. No modelo (1), as nossas equações usam passos de tempo $\Delta t = 1$, representando um dia. Por causa disso o Δt fica oculto. Apresentamos resultados de experimentos numéricos, onde analisamos o comportamento do modelo, utilizando diferentes valores para Δt .

3. RESULTADOS E DISCUSSÃO

Nas próximas seções, apresentamos os resultados dos experimentos numéricos realizados. Nosso objetivo é tentar diminuir o máximo que conseguirmos o pico no número de infectados, removendo as arestas do grafo de infecção, mas sem deixá-lo desconexo. Na Seção 3.1, apresentamos resultados de experimentos onde as remoções são feitas manualmente, seguindo algum critério. Na Seção 3.2, apresentamos os resultados com remoções baseadas em algoritmos de Busca em Largura e Busca em Profundidade. Na Seção 3.3 apresentamos uma heurística para encontrar novas árvores geradoras e na Seção 3.4 apresentamos uma análise de convergência do método de Euler com diferentes valores de Δt .

3.1. Tentativas iniciais

Em [Franco 2020], percebe-se através dos experimentos realizados, que a topologia do grafo altera o comportamento do espalhamento de uma doença. As topologias *linha* e *cíclica* mostraram-se as topologias com os melhores resultados, isto é, a curva de infectados sobe de forma suave, estabiliza e depois desce da mesma forma. As curvas de suscetíveis e recuperados também se comportam de forma suave, descendo e subindo, respectivamente. Dito isso, a primeira tentativa foi feita mudando uma seção do grafo (18 vértices no total). A ideia foi construir dois ciclos a partir dos vértices da zona sul do Rio de Janeiro, usando o bairro Botafogo como ligação entre os ciclos. Parte dos vértices restantes (que não fazem parte dos ciclos) formam novos caminhos, onde uma ponta do caminho faz parte do ciclo e a outra pode ou não fazer parte dele, formando novos ciclos ou caminhos sem saída, respectivamente, mantendo sempre o grafo conexo. A Figura 10 apresenta esta modificação.

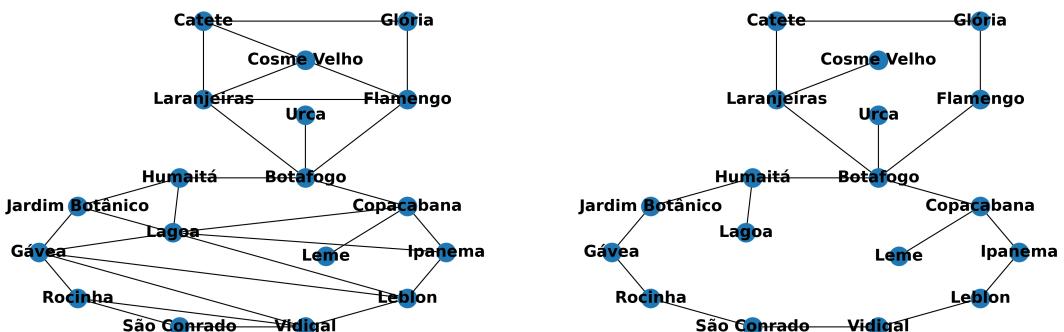


Figura 10. Grafo original da zona sul do RJ (à esquerda) e grafo com ciclos e caminhos (à direita), contendo os mesmos vértices.

Utilizamos o subgrafo da zona sul do Rio com o objetivo de ter mais controle sobre a forma do grafo e como mudanças nele alteram os resultados. Nele o pico de infectados ficou em torno de 125.000 pessoas. Com a remoção destas arestas o pico de infectados diminuiu. Dependendo do bairro escolhido como início do espalhamento, houve uma diminuição no pico de até 4000 pessoas, como no caso do início no bairro de Copacabana ou na Gávea, mas iniciando-se em outros bairros essa diminuição foi bem menor, como no caso do início nos bairros do Flamengo ou Cosme Velho, onde o pico diminuiu em menos de 800 pessoas.

Um outro experimento foi realizado sobre a zona sul, transformando-a em um grafo minimal, ou seja, removendo o máximo de arestas possíveis deixando o grafo conexo. A remoção de qualquer outra aresta tornaria o grafo desconexo. Para fazer isso, basta remover uma aresta de cada um dos dois ciclos existentes na Figura 10 (à direita). Dependendo de quais arestas são removidas e o vértice de início do espalhamento, o pico de infectados foi diferente. O menor pico que conseguimos alterando essas duas variáveis foi de 68.000 pessoas. Ele foi obtido no grafo em que as arestas Copacabana-Botafogo e Flamengo-Botafogo foram removidas e com o início do espalhamento no vértice do Flamengo, apresentado na Figura 11.

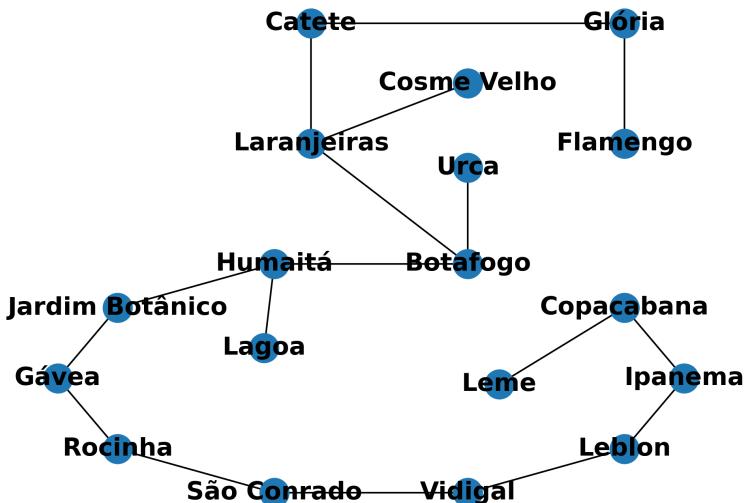


Figura 11. Grafo minimamente conexo da zona sul do RJ.

O menor pico aconteceu nessas circunstâncias pela combinação de dois fatores. O primeiro foi a escolha das arestas a se remover, envolvendo Copacabana, Botafogo e Flamengo, os três bairros mais populosos da zona sul. A remoção das suas conexões faz com que a infecção tenha que percorrer um longo caminho, passando por bairros menos populosos, fazendo com que os suscetíveis se infectem mais lentamente. O segundo fator foi o início do espalhamento no bairro Flamengo, que dificultou mais ainda o espalhamento, fazendo com que infectados tivessem que começar na ponta do grafo, lentamente caminhando de bairro em bairro e infectando menos pessoas a cada dia. Foi observado então que quanto mais linear é o grafo minimal melhor.

Observando o gráfico da Figura 12 podemos ver dois picos diferentes, nos dias 35

e 60, que acontecem por causa da remoção dessas arestas. O primeiro pico acontece depois dos infectados chegarem em Botafogo nos primeiros 10 dias e começarem a infectar os suscetíveis dele. Logo depois, temos o maior pico, que acontece logo depois de um infectado chegar em Copacabana, rapidamente infectando vários suscetíveis e levando a curva de infectados para cima novamente.

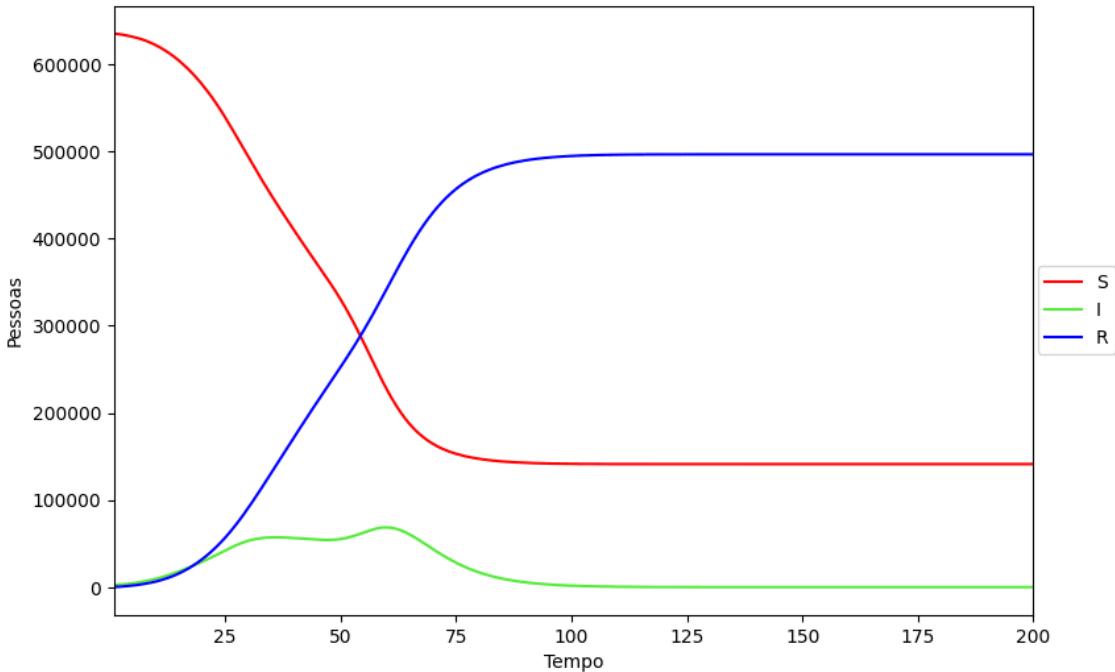


Figura 12. Gráfico de Infecção ao longo de 200 dias no grafo minimal da zona sul.

Utilizamos agora um outro método para alterar a topologia do grafo. A ideia é isolar os vértices mais populosos do grafo quando se percebe um aumento significativo no número de infectados durante a simulação. O isolamento é realizado de forma cíclica entre o estado isolado e normal, tendo a duração de um ciclo definida inicialmente pelo número de dias até o pico de infectados do grafo original dividido por 10 e o período total durando até o dia do antigo pico. Desta vez, o método foi realizado sobre o grafo completo do Rio de Janeiro, com ciclos de 7 dias, a partir do dia 18 (onde houve diferença significativa de infectados de um dia para o outro) até o dia 73 (pico original). Com a utilização desse método o pico diminuiu de 810.000 para cerca de 660.000. Mudar a duração do ciclo, tanto para mais quanto para menos, parece aumentar o pico de infectados, assim como alterar o dia de começo ou de término dos isolamentos cílicos.

3.2. Obtenção de árvores geradoras através de buscas

Até agora fizemos apenas remoções manuais de arestas, baseadas em alguma ideia criada por nós, mas também podemos removê-las com base em algoritmos, como os de busca em largura ou em profundidade, apresentados na Seção 2.4. Utilizando-se dessas novas estratégias que encontram um subgrafo conexo do grafo original obtivemos resultados interessantes. Para os grafos gerados a partir de árvores em largura obtivemos o gráfico da Figura 13. Nele os pontos em vermelho equivalem ao pico de infectados no grafo não

modificado do RJ com início do espalhamento no bairro descrito no eixo x . Os pontos verdes equivalem ao pico de infectados no grafo modificado pelo algoritmo de busca em largura. Nem todos os bairros estão explicitamente nomeados.

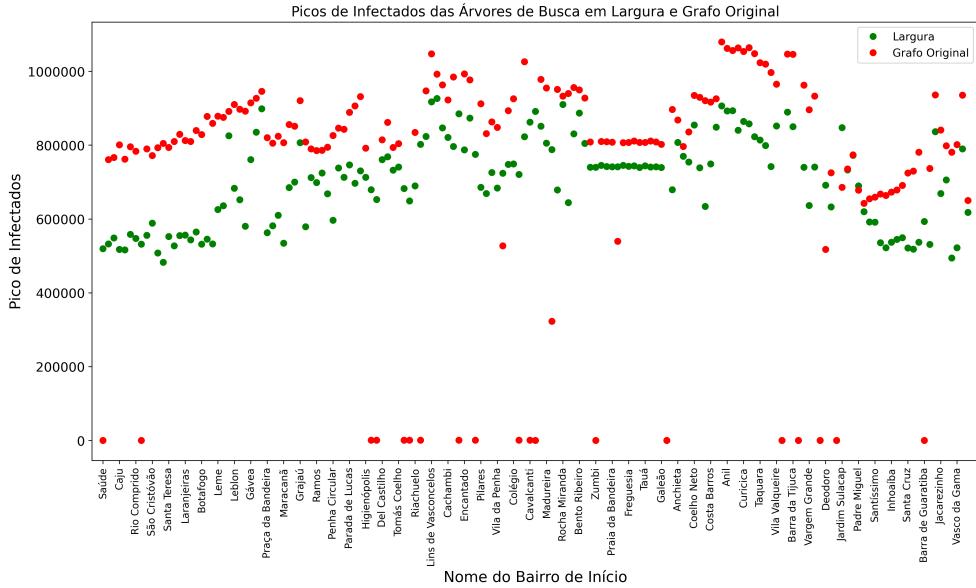


Figura 13. Pico de Infectados das árvores de largura geradas a partir de cada $v \in V(G)$.

Os valores, quando comparados com o grafo original mostram que em geral houve diminuição no pico de infectados. Já as árvores de busca em profundidade foram testadas com 400 dias, pois o espalhamento da doença não terminava antes dos 200 dias. A Figura 14 apresenta a comparação entre o pico de infectados do grafo original e o grafo modificado pelo algoritmo de busca em profundidade.

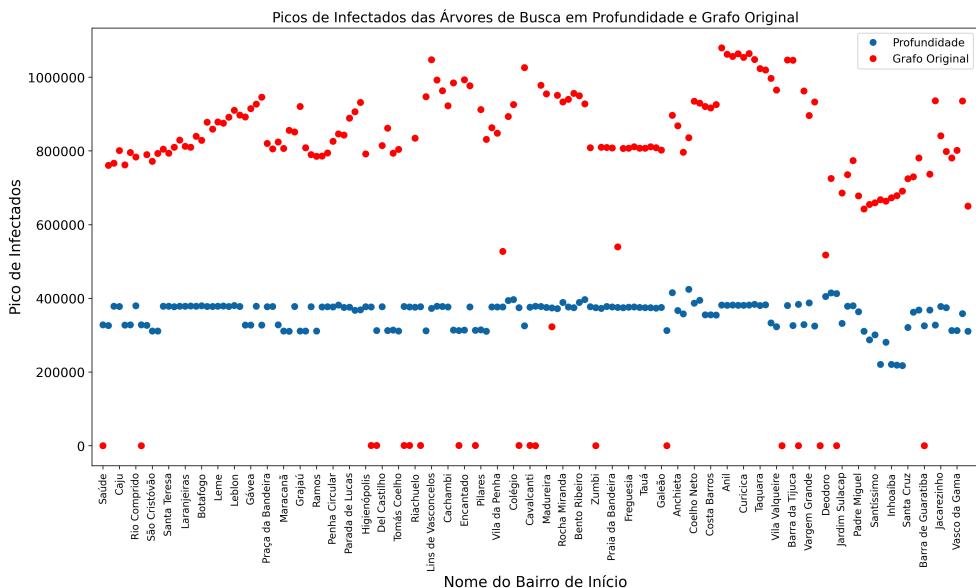


Figura 14. Pico de Infectados ($t_F = 400$) das árvores de profundidade geradas a partir de cada $v \in V(G)$.

Os experimentos nos indicam que manter o grafo de infecção original proporciona, para quase todos os casos, um pico de pessoas infectadas maior que as topologias das árvores geradoras testadas. As árvores de busca em profundidade possuem grande destaque no que diz respeito ao número de pessoas infectadas no pico, tal número permanece muito próximo de uma constante. No entanto, vale a pena citar que em alguns casos, o grafo de infecção original é a melhor opção de topologia pois a doença não conseguiu se espalhar (a distribuição das pessoas da forma como foi considerada no modelo implicou no número esperado de infectados igual a 0 (zero) para todo vértice logo no início da simulação). Isso acontece quando o espalhamento começa nos bairros Saúde, Cavalcanti, Zumbi ou Barra de Guaratiba, por exemplo. Temos apenas um caso em que a melhor opção é o grafo de infecção original e o pico de infectados não é igual a zero para qualquer topologia considerada, Vaz Lobo, imediatamente após Madureira. A Figura 15 apresenta a comparação dos picos nas árvores de busca em largura, de busca de profundidade e do grafo original.

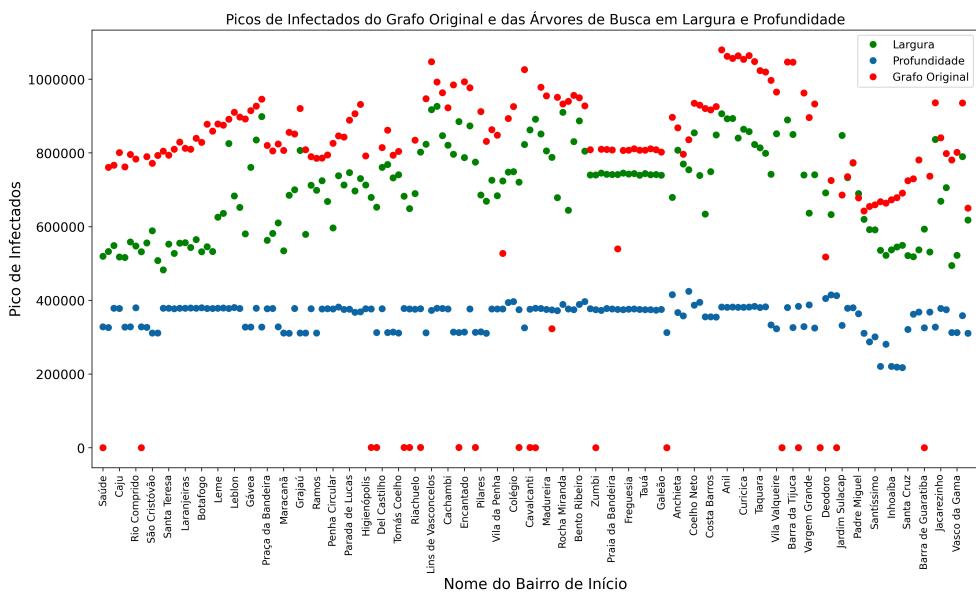


Figura 15. Comparação do Pico de Infectados das árvores de largura e profundidade e grafo original.

Podemos ainda analisar o comportamento das árvores de busca em largura e busca em profundidade ao longo do tempo em cada árvore. Nos experimentos realizados, há uma diferença no comportamento geral entre as árvores de busca em largura e em profundidade. O espalhamento da doença é mais lento nas árvores de busca em profundidade. Para ilustrar este comportamento, a Figura 16 apresenta os números de suscetíveis, infectados e recuperados ao longo do tempo das árvores de busca em largura e busca em profundidade a partir do vértice do bairro do Flamengo. O experimento também considera que o espalhamento da doença inicia no bairro do Flamengo. Todas as árvores de busca em largura tiveram um comportamento similar ao gráfico da esquerda, com exceção daqueles que a doença não se desenvolveu. Já as árvores de busca em profundidade, em sua maioria, tiveram um comportamento similar ao do gráfico da direita, com picos menores e curvas de infectados mais suaves e esticadas ao longo do tempo.

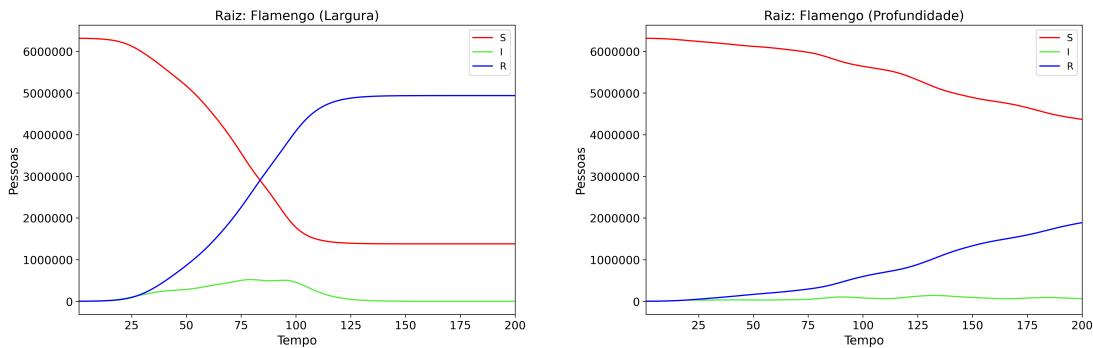


Figura 16. Gráficos de Infecção das árvores de busca em largura (à direita) e busca em profundidade (à esquerda), com início no bairro do Flamengo.

A Figura 17 apresenta o gráfico do espalhamento da doença da árvore de busca em profundidade do bairro do Flamengo (Figura 16 à direita) em um intervalo de 400 dias.

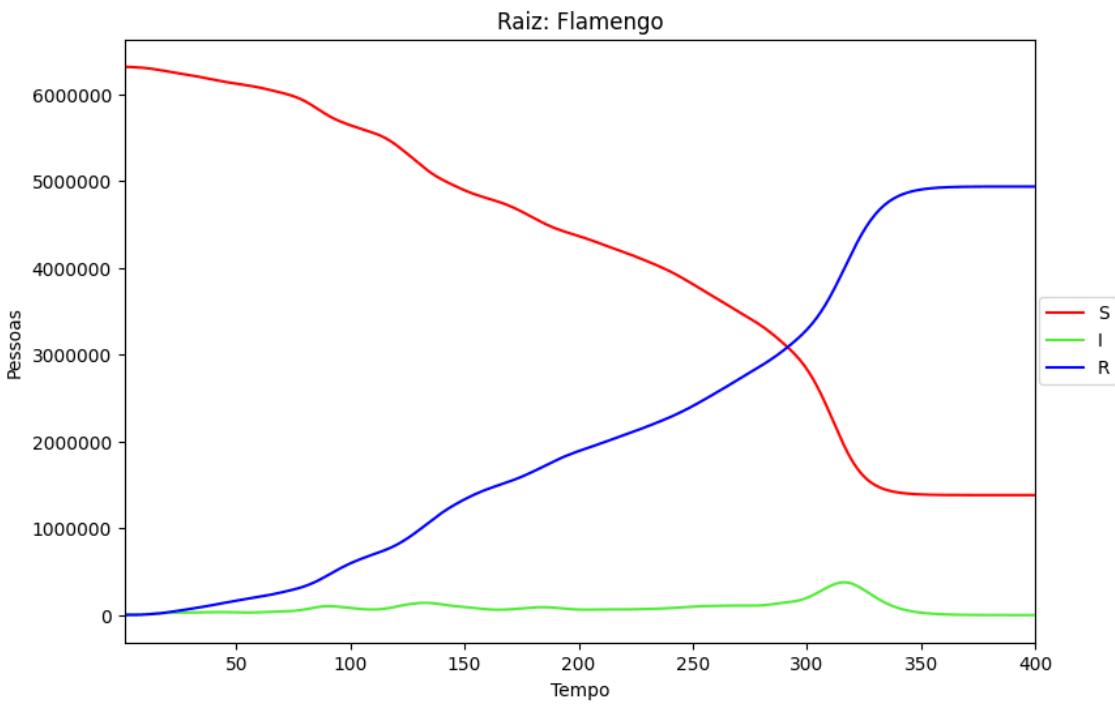


Figura 17. Gráfico de Infecção ao longo de 400 dias da árvores de profundidade gerada a partir do vértice Flamengo.

É possível observar que a infecção continuou se espalhando na população até mais ou menos o dia 390, com um número de infectados muito pequeno. O pico da doença aconteceu no dia 316, bem depois do que acontece em árvores de largura ou no grafo original da Figura 6. Analisando os gráficos de 200 dias de outras árvores de profundidade é possível perceber que em todas elas o espalhamento não termina antes desses 200 dias.

Repetimos os experimentos para o grafo de infecção da cidade de Florianópolis. Para este caso, aplicamos o modelo sobre o grafo de infecção original apenas uma vez, já que são conhecidos os números reais de suscetíveis, infectados, e recuperados de cada

vértice em um determinado dia (dados reais) obtidos em [Ataide 2022]. Também aplicamos sobre as árvores de busca em largura e em profundidade, onde cada grupo de bairros é a raiz de uma dessas árvores, totalizando 16×2 árvores geradoras. Os resultados relacionados ao número de infectados no pico podem ser observados na Figura 18. A linha horizontal que cruza esta figura está sobre o número 83.271, o número de infectados no pico considerando o grafo de infecção original.

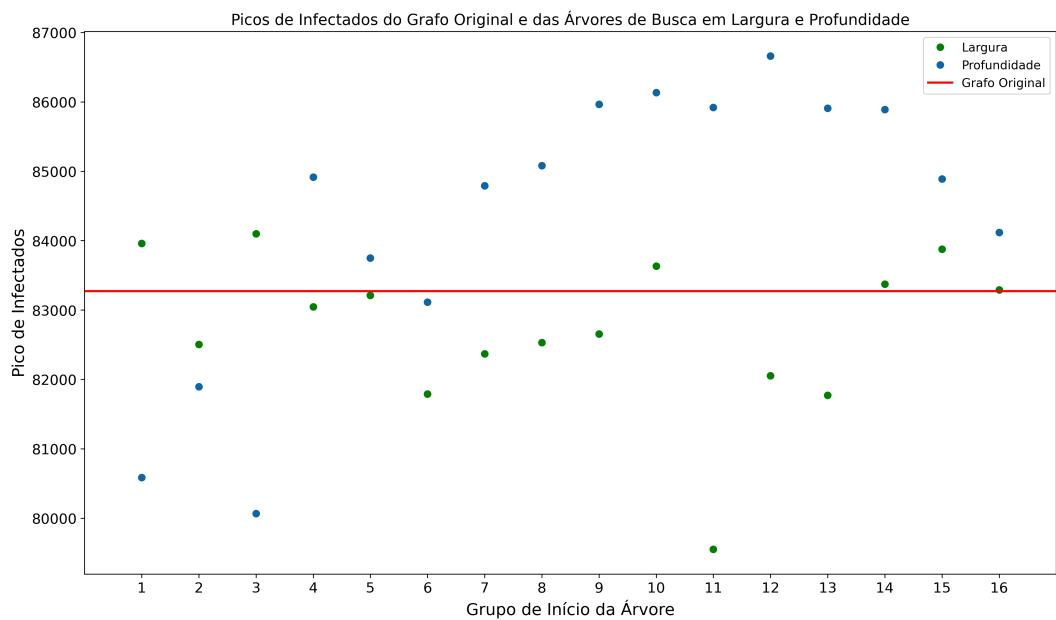


Figura 18. Florianópolis - Observe duas curvas para os picos de infectados para cada uma das árvores de busca em largura e em profundidade, além do pico de infectados considerando o grafo de infecção original.

Neste caso, os experimentos nos indicam que há muitas árvores geradoras (daque-las consideradas) cujo número de infectados no pico é maior que o número de infectados no pico do grafo de infecção original. A maioria das árvores de busca em profundidade possuem pico de infectados maior que o pico de infectados do grafo de infecção original. Enquanto que a maioria das árvores de busca em largura possuem pico de infectados menor. Das árvores geradoras consideradas, aquela que obteve o menor número de infectados no pico foi uma árvore de busca em largura com raiz no Grupo 11, composto pelos bairros Rio Tavares e Costeira do Pirajubaé. Todos os grupos são apresentados na Seção 2.2.

Na próxima seção, descrevemos uma heurística que, a partir de uma árvore geradora (em largura ou em profundidade), passa por outras árvores geradoras do grafo de infecção, *vizinhas* da árvore de partida, e destaca aquela com o menor número de infectados.

3.3. Uma heurística que encontra novas árvores geradoras

Nesta seção, descrevemos um procedimento que recebe uma árvore geradora T de um grafo de infecção G e devolve uma lista de outras árvores geradoras de G , caso o conjunto de arestas de G seja diferente do conjunto de arestas de T . Sejam $E(G)$ e $E(T)$ os

conjuntos de arestas de G e de T , respectivamente. Definimos uma *árvore vizinha* T' de T se T' difere de T em uma aresta. Dessa forma, podemos construir uma lista com árvores vizinhas de T da seguinte maneira. Seja e uma aresta de $E(G) - E(T)$, isto é, uma aresta de G que não está na árvore T . Ao adicionar e em T , criamos um grafo com exatamente um ciclo ($T + e$). Quase todas as arestas deste ciclo estão na árvore T , exceto por e . Considerando e_1, e_2, \dots, e_k todas as arestas do ciclo que estão na árvore T , ao removermos exatamente uma aresta e_i , obtemos uma nova árvore (geradora de G) vizinha de T . Dessa forma, construímos k árvores vizinhas de T ao remover uma aresta e_i por vez. Para construir uma lista L com todas as árvores vizinhas de T , basta repetirmos o procedimento descrito anteriormente para cada aresta em $E(G) - E(T)$. O modelo epidemiológico pode ser executado considerando todas as árvores vizinhas de uma árvore geradora T , encontrando assim, um mínimo local.

O tempo de execução da heurística por árvore geradora é da ordem de $O(nm)$ sendo n e m os números de vértices e de arestas de G , respectivamente. Isso porque para cada uma das $m - n + 1$ ($O(m)$) arestas de $E(G) - E(T)$, adicionar e em T e encontrar o ciclo consome tempo $O(n)$. Para encontrar o ciclo, basta seguirmos pelos pais das pontas de e na árvore T até chegarmos em um ancestral comum. Então, se a lista de árvores vizinhas de T possui tamanho k , encontrar um mínimo local considerando uma árvore T leva tempo $O(knm)$. No pior caso, teremos $k = O(nm)$. Isso porque, para cada $O(m)$ aresta inserida em T , podemos ter ciclos de tamanho $O(n)$ e assim, podemos criar $O(n)$ árvores vizinhas para cada inserção de aresta em T . Logo, o tempo de execução no pior caso é $O(n^2m^2)$. Para o nosso caso específico, essa heurística é executada em tempo $2n = O(n)$, uma execução para cada árvore de busca em largura e em profundidade com raiz em cada vértice do grafo de infecção G com n vértices. Então, para o nosso caso específico, o tempo de execução do procedimento total fica igual a $O(n^3m^2)$. Note que a lista de árvores vizinhas não precisa ser construída explicitamente pois podemos manter apenas aquela que possui o melhor resultado. Então, o espaço consumido para guardar uma árvore geradora vizinha é suficiente, ou seja, $O(n)$.

Em seguida apresentamos os resultados experimentais da aplicação dessa heurística sobre algumas árvores geradoras (de busca em largura e em profundidade) do grafo de infecção da cidade de Florianópolis. Os resultados estão em gráficos *boxplot*. Uma linha horizontal no gráfico está sobre o número 83.271, o pico da curva de infectados do grafo de infecção original. Cada *boxplot* corresponde aos resultados do pico da curva de infectados das árvores vizinhas de uma árvore geradora com raiz em um vértice do grafo de infecção original. Podemos ver na Figura 19 que a maioria das árvores vizinhas de árvores de busca em profundidade possuem resultados piores que o resultado do grafo de infecção original. Já observando a Figura 20, essa conclusão se inverte. A maioria das árvores vizinhas de árvores de busca em largura possuem resultados melhores que o resultado do grafo original. O melhor resultado foi obtido através de uma árvore vizinha de uma árvore de busca em largura com raiz no Grupo 7. Neste caso, o pico atingiu 70.647, ou seja, o pico de infectados da árvore vizinha é 15,16% menor que o pico de infectados do grafo original. É importante também dizer que os experimentos indicam que há muitas árvores vizinhas de árvores geradoras que possuem resultados piores quando comparados com o resultado do grafo de infecção original. Há árvores vizinhas de árvores de busca em profundidade com mais de 6% de infectados no pico (Grupos 7 e 15) quando comparados com o pico do grafo de infecção original. Note na Figura 20 que

pelo menos $\frac{3}{4}$ das árvores vizinhas de árvores de busca em largura com raiz nos Grupos 4, 6, 7, 11 e 12 possuem resultados melhores que o resultado do grafo de infecção original. Em contrapartida, note na Figura 19 que nem mesmo $\frac{1}{4}$ das árvores vizinhas de árvores de busca em profundidade com raiz nos Grupos 12 e 13 possuem resultados melhores que o resultado do grafo de infecção original.

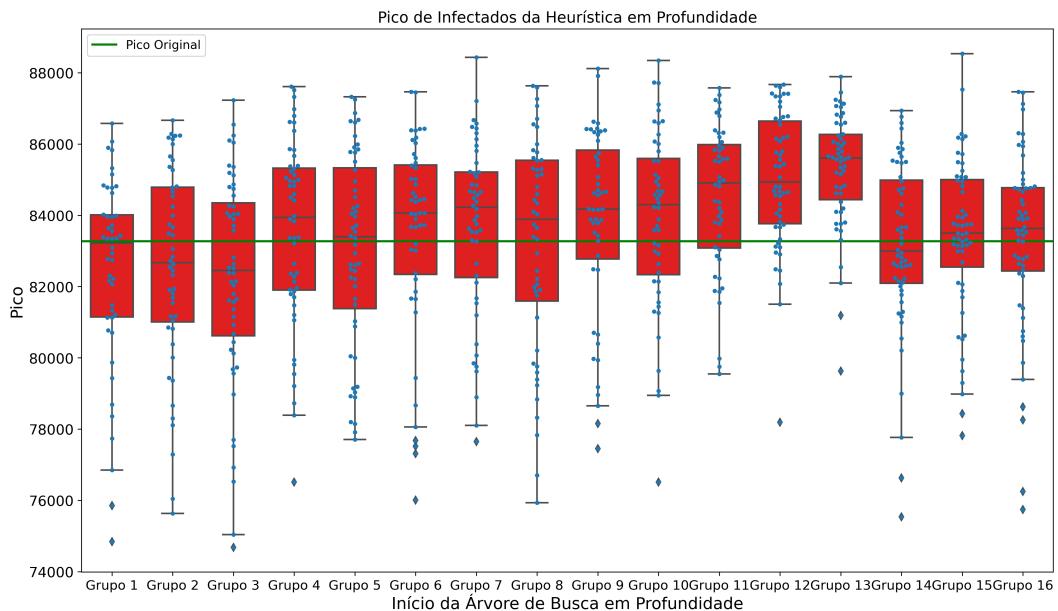


Figura 19. Florianópolis - Boxplot com os resultados do pico de infectados da vizinhança sobre árvores de busca em profundidade.

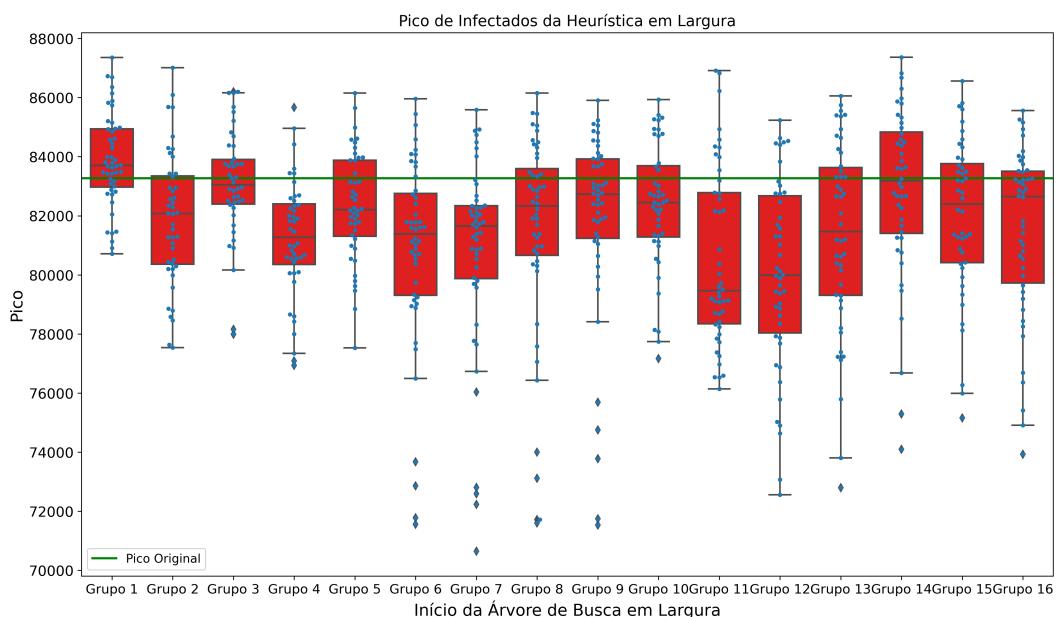


Figura 20. Florianópolis - Boxplot com os resultados do pico de infectados da vizinhança sobre árvores de busca em largura.

3.4. Simulação numérica com o método de Euler

Nesta seção, apresentamos uma análise de convergência do modelo epidemiológico (2), que corresponde a discretização do sistema de EDOs (3) por meio do método de Euler. A Figura 21 apresenta a curva de infectados com diferentes Δt em ordem decrescente.

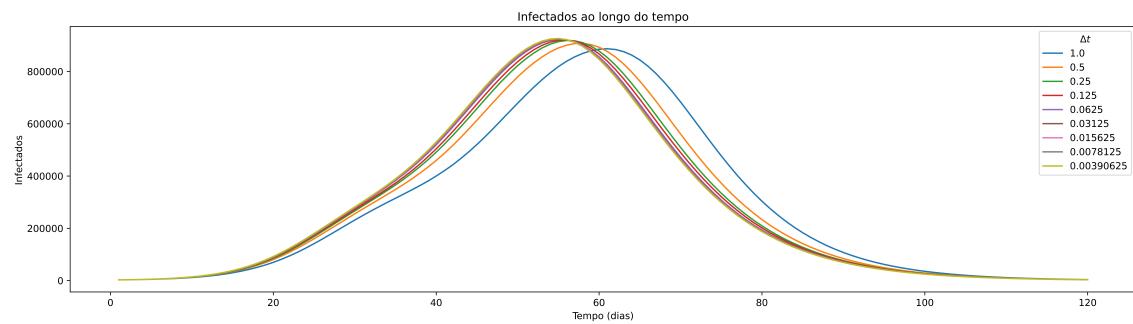


Figura 21. Curva dos infectados ao longo do tempo com diferentes Δt

Podemos ainda observar que não só o número de infectados no pico é alterado, mas também o dia em que ele acontece. Quanto menor o Δt , mais as curvas se aproximam entre si. A Figura 22 apresenta uma aproximação da Figura 21 na região dos picos.

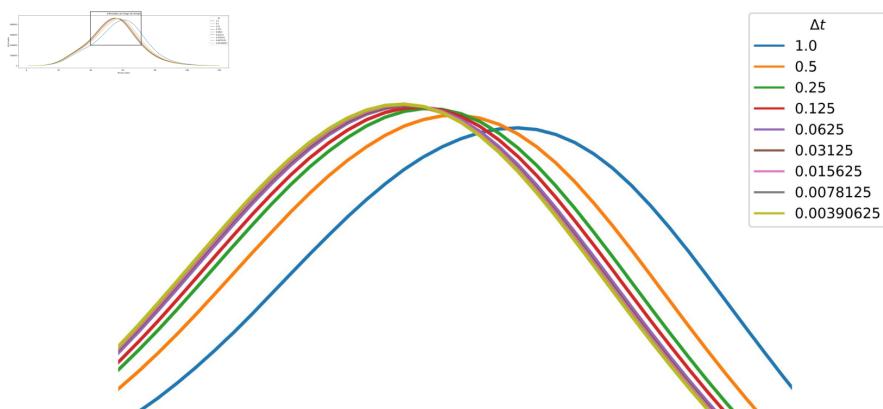


Figura 22. Aproximação da região dos picos de infectados com diferentes Δt .

Já a Figura 23 apresenta o número de infectados durante o pico de infecção com diferentes valores de Δt . O valor ao lado dos pontos apresenta a diferença percentual quando comparado com o número de infectados obtido com o Δt anterior.

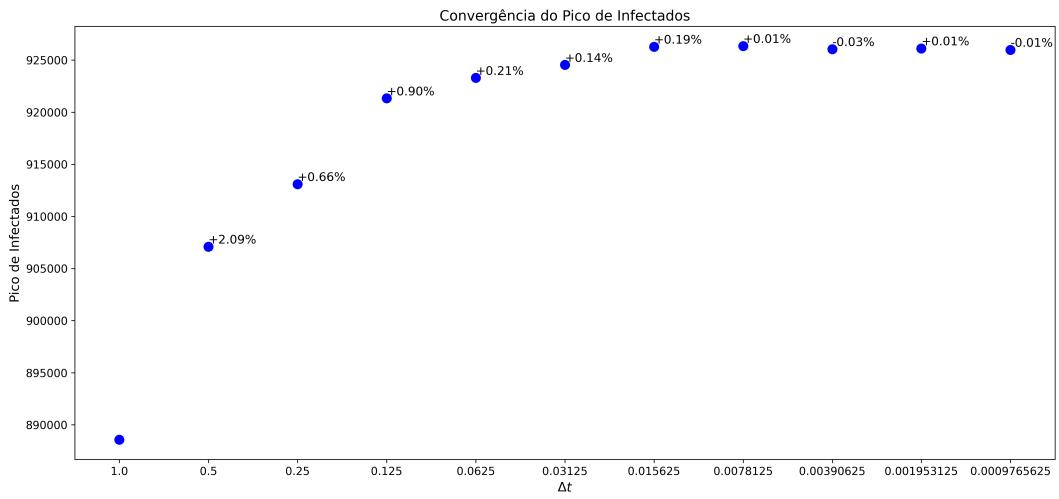


Figura 23. Picos de infectados com diferentes Δt .

O último experimento, analisa a influência no parâmetro α (fator de distanciamento social). Para isso variamos o parâmetro α , com $\Delta t = 0.0625$. A Figura 24 apresenta o número de infectados no pico da infecção para diferentes fatores de distanciamento social. Podemos observar, conforme o esperado, a diminuição no número de infectados, entretanto a doença ainda se espalha pois mesmo respeitando o distanciamento algumas pessoas precisam se movimentar.

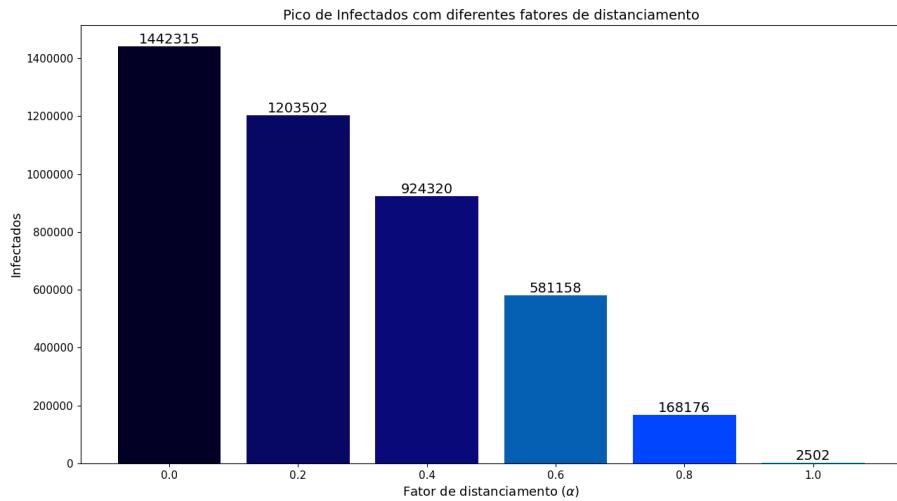


Figura 24. Picos de infectados com diferentes valores para o fator distanciamento social α .

4. CONSIDERAÇÕES FINAIS

Neste trabalho, aplicamos um modelo epidemiológico sobre grafos de infecção. Utilizamos dois grafos de infecção correspondentes a duas cidades do Brasil: Rio de Janeiro e Florianópolis. Cada bairro (ou grupo de bairros para o caso de Florianópolis) é representado como um vértice neste grafo e bairros vizinhos (ou grupos de bairros vizinhos)

são conectados por uma aresta do grafo de infecção. Foram utilizados dados conhecidos e reais sobre o número de pessoas que vivem nos bairros. Para o caso de Florianópolis, utilizamos ainda dados reais do número de pessoas suscetíveis, infectadas e recuperadas, considerando a doença COVID-19. Apresentamos uma heurística que encontra mínimo local dada uma definição de vizinhança de árvore geradora. Foram apresentados os resultados experimentais obtidos pela aplicação do modelo epidemiológico sobre algumas árvores geradoras dos grafos de infecção das duas cidades.

Realizamos diversas alterações topológicas, obtendo árvores geradoras por vários métodos diferentes. A partir dos resultados, conseguimos concluir que na maioria dos experimentos realizados, a restrição da topologia nos leva a um menor pico de infecção no grafo. Isso acontece em decorrência da restrição de movimento que essas mudanças topológicas proporcionam. A movimentação das pessoas é dificultada e os experimentos ilustram que o espalhamento da doença no grafo é reduzido. Com essas restrições, pode ser impossível a movimentação direta de pessoas infectadas entre vértices adjacentes do grafo original. Com isso, a infecção pode levar mais tempo para atingir tais vértices. Vale ressaltar a importância de controlar o espalhamento de uma doença. Uma menor quantidade de pessoas infectadas simultaneamente implica em menor ocupação em hospitais e em leitos de UTIs, o que nos leva a possibilidade de tratar eficientemente cada um dos pacientes de acordo com a gravidade dos sintomas.

Nosso último experimento ressaltou a importância de conhecer o modelo e os erros que uma discretização pode trazer. O erro vai se acumulando a cada passo que realizamos e se não o levarmos em conta podemos ter previsões erradas e tomar decisões equivocadamente.

A iniciação científica proporcionou um grande aprendizado em diversas áreas de conhecimento como grafos, programação, cálculo e escrita. Aprimorou também o meu pensamento crítico por estar constantemente em contato com o método científico. Uma das maiores contribuições dela foi *aprender a aprender*.

4.1. Participações em eventos e publicações

Parte desse trabalho será apresentado no LV Simpósio Brasileiro de Pesquisa Operacional 2023 (SBPO), que ocorre em novembro de 2023, em São José do Campos - SP. O artigo que será publicado nos Anais do evento, apresenta os resultados das árvores geradoras obtidas através de busca e da heurística utilizada na Seção 3.3. Além disso, foi apresentado no Encontro Acadêmico de Matemática (EAMat) da UFSC, realizado em agosto, os resultados obtidos pela simulação numérica do modelo com o método de Euler.

5. AGRADECIMENTOS

Este trabalho recebeu apoio financeiro da Fundação de Amparo à Pesquisa e Inovação do Estado de Santa Catarina - FAPESC n. 2021TR000436, do Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq) e da Universidade Federal de Santa Catarina - UFSC.

Referências

Ataide, W. C. (2022). Minimização do contágio em uma topologia com Programação Matemática. Trabalho de conclusão de curso, Universidade Federal de Santa Catarina. Acessado: 02-06-2022.

- Barbosa, W. F., da Costa Moreira, E. B., de Araújo, J. M., Pazin-Filho, A., and Bueno, C. D. F. (2021). Modelo seir para avaliação do comportamento da pandemia de covid-19 em marabá-pa. *Revista de Medicina*, 100(4):322–328.
- Burden, R. L. and Faires, J. D. (2011). *Numerical analysis*. Cengage learning, 9th edition.
- Dasgupta, S., Papadimitriou, C. H., and Vazirani, U. V. (2008). *Algorithms*. McGraw-Hill Higher Education New York.
- Feofiloff, P., Kohayakawa, Y., and Wakabayashi, Y. (2011). Uma introdução sucinta à teoria dos grafos.
- Franco, Á. J. P. (2020). Epidemic model with restricted circulation and social distancing on some network topologies. In *International Conference on Cellular Automata for Research and Industry*, pages 261–264. Springer.
- Hethcote, H. W. (1989). Three basic epidemiological models. In *Applied Mathematical Ecology*, pages 119–144. Springer.
- Keeling, M. J. and Eames, K. T. (2005). Networks and epidemic models. *Journal of the royal society interface*, 2(4):295–307.
- Linge, S. and Langtangen, H. P. (2020). *Programming for computations-Python: A gentle introduction to numerical simulations with Python 3.6*. Springer Nature.
- Manrique-Abril, F. G., Agudelo-Calderon, C. A., González-Chordá, V. M., Gutiérrez-Lemes, O., Téllez-Piñerez, C. F., and Herrera-Amaya, G. (2020). Modelo sir de la pandemia de covid-19 en colombia. *Revista de Salud Pública*, 22:123–131.
- Rio, D. (2010a). População residente, por grupos de idade e sexo, segundo as Áreas de planejamento (ap), regiões administrativas (ra) e bairros. *Prefeitura da Cidade do Rio de Janeiro. Tabela*, 2974.
- Rio, D. (2010b). População residente, por idade e por grupos de idade, segundo as Áreas de planejamento (ap), regiões administrativas (ra) e bairros. *Prefeitura da Cidade do Rio de Janeiro. Tabela*, 2973.
- Rio, D. (2010c). População residente por sexo e razão de sexos, segundo as Áreas de planejamento (ap), regiões de planejamento (rp), regiões administrativas (ra) e bairros no município do rio de janeiro. *Prefeitura da Cidade do Rio de Janeiro. Tabela*, 3163.
- Visbal, J. H. W. and Pedraza, M. C. C. (2020). Aproximación matemática del modelo epidemiológico sir para la comprensión de las medidas de contención contra la covid-19. *Rev. esp. salud pública*.
- White, S. H., Del Rey, A. M., and Sánchez, G. R. (2007). Modeling epidemics using cellular automata. *Applied Mathematics and Computation*, 186(1):193–202.