



**UNIVERSIDADE FEDERAL DE CAMPINA GRANDE
CENTRO DE ENGENHARIA ELÉTRICA E INFORMÁTICA
UNIDADE ACADÊMICA DE SISTEMAS E COMPUTAÇÃO
CURSO DE BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO**

LUCAS FERNANDES DE OLIVEIRA

**ALGORITMOS PARA DETECÇÃO E VISUALIZAÇÃO DE
AGLOMERADOS DE ÔNIBUS**

CAMPINA GRANDE - PB

2019

LUCAS FERNANDES DE OLIVEIRA

**ALGORITMOS PARA DETECÇÃO E VISUALIZAÇÃO DE
AGLOMERADOS DE ÔNIBUS**

**Trabalho de Conclusão Curso
apresentado ao Curso Bacharelado em
Ciência da Computação do Centro de
Engenharia Elétrica e Informática da
Universidade Federal de Campina
Grande, como requisito parcial para
obtenção do título de Bacharel em Ciência
da Computação.**

Orientador: Professor Dr. Carlos Eduardo Santos Pires.

CAMPINA GRANDE - PB

2019



048a Oliveira, Lucas Fernandes de.
Algoritmos para detecção e visualização de aglomerados de ônibus. / Lucas Fernandes de Oliveira. - 2019.

13 f.

Orientador: Prof. Dr. Carlos Eduardo Santos Pires.
Trabalho de Conclusão de Curso - Artigo (Curso de Bacharelado em Ciência da Computação) - Universidade Federal de Campina Grande; Centro de Engenharia Elétrica e Informática.

1. Algoritmos de detecção. 2. Transporte público. 3. Siperlotação de ônibus. 4. Headway. 5. Sistema de monitoramento de ônibus. I. Pires, Carlos Eduardo Santos. II. Título.

CDU:004(045)

Elaboração da Ficha Catalográfica:

Johnny Rodrigues Barbosa
Bibliotecário-Documentalista
CRB-15/626

LUCAS FERNANDES DE OLIVEIRA

**ALGORITMOS PARA DETECÇÃO E VISUALIZAÇÃO DE
AGLOMERADOS DE ÔNIBUS**

**Trabalho de Conclusão Curso
apresentado ao Curso Bacharelado em
Ciência da Computação do Centro de
Engenharia Elétrica e Informática da
Universidade Federal de Campina
Grande, como requisito parcial para
obtenção do título de Bacharel em Ciência
da Computação.**

BANCA EXAMINADORA:

**Professor Dr. Carlos Eduardo Santos Pires
Orientador – UASC/CEEI/UFCG**

**Professor Dr. Nazareno Andrade
Examinador – UASC/CEEI/UFCG**

**Professor Dr. Tiago Lima Massoni
Disciplina TCC – UASC/CEEI/UFCG**

Trabalho aprovado em: 02 de julho de 2019.

CAMPINA GRANDE - PB

ALGORITMOS PARA DETECÇÃO E VISUALIZAÇÃO DE AGLOMERADOS DE ÔNIBUS

Lucas Fernandes de Oliveira
Orientador: Prof. Dr. Carlos Eduardo Santos Pires
Departamento de Sistemas e Computação
Universidade Federal de Campina Grande
Campina Grande, Paraíba, Brasil
lucas.oliveira@ccc.ufcg.edu.br

RESUMO

Atrasos nas viagens e superlotação de ônibus são algumas das insatisfações diárias de quem faz uso de transporte público no Brasil. Uma destas causas são os aglomerados de ônibus, os quais ocorrem quando dois ou mais ônibus executam a mesma rota juntos, ocasionando aumento no tempo de espera e concentração de passageiros em paradas. Com foco neste problema, foram implementados algoritmos para um sistema de monitoramento de ônibus ser capaz de detectar e exibir aglomerados de ônibus em tempo real. A aplicação utiliza dados de GPS dos ônibus em tempo real e, para a detecção dos aglomerados, os algoritmos baseiam-se em métricas de distância e headway. A aplicação modificada apresenta visualmente, com clareza, as ocorrências dos aglomerados de ônibus. A avaliação do algoritmo de headway apresentou resultados eficazes, com desvio mínimo para possíveis casos falsos positivos. A algoritmo de distância, apesar de exibir resultados eficazes na sua avaliação, apresentou uma quantidade significativa de resultados contraditórios à definição de aglomerados de ônibus. Em comparação dos dois algoritmos, o de *headway* mostrou-se mais confiável.

PALAVRAS-CHAVE

transporte público, aglomerados, ônibus, *headway*, tempo real, qualidade de serviço.

CCS CONCEPTS

Data Analytics and Data Visualization; Real-time Databases; Spatial Databases and Geographical Information Systems

1 INTRODUÇÃO

O Brasil teve o seu processo de industrialização ocorrendo de forma tardia e em uma velocidade bastante elevada, assim como aconteceu na ampla maioria dos países subdesenvolvidos e em desenvolvimento. O que sempre se viu na história das cidades brasileiras foi uma grande massa de trabalhadores deslocando-se através de ônibus lotados em grandes distâncias (UOL, MUNDO EDUCAÇÃO). Apesar de ser um serviço acessível a todos, que contribui na redução de congestionamentos, emissão de gases e consumo de energia, ainda é algo que precisa de mais planejamento para atender às pessoas com a qualidade de vida.

Em um ambiente estocástico, como é o trânsito nas cidades grandes, atrasos nos cronogramas dos ônibus são quase que inevitáveis. Uma das principais preocupações das empresas de transporte, os aglomerados de ônibus (em inglês, *bus bunching*), ocorre quando vários ônibus executam a mesma rota juntos, chegando em um espaço de tempo muito curto na mesma parada (FENG & FIGLIOZZI, 2011) (DIAB et al., 2016). Esse problema pode se relacionar diretamente com inúmeros fatores como horários de pico, acidentes de trânsito, poucos recursos para suprir grandes demandas, desvio do horário programado e também cortes de veículos e rotas.

Devido à complexidade da ocorrência dos aglomerados, sendo o mesmo um problema cíclico, é de interesse das empresas de transporte resolver/evitar a ocorrência dos aglomerados. Por exemplo, um pequeno atraso ou congestionamento faz aumentar a quantidade de pessoas nas paradas de ônibus. Um ônibus, por sua vez, permanece mais tempo na parada pegando todos esses passageiros e, enquanto este transita superlotado, o próximo ônibus seguirá rápido e vazio. Eventualmente, um ônibus irá alcançar o outro, assim formando o aglomerado (MOREIRA-MATIAS et al., 2016) (MUNIZAGA et al., 2018). Esse cenário provoca um acréscimo no tempo de espera dos passageiros

por um ônibus em outros pontos e uma distribuição irregular nos veículos, evidenciando um desperdício de recursos e refletindo um serviço sem credibilidade à população.

A possibilidade de detectar a ocorrência de aglomerados em tempo real pode fornecer às empresas de ônibus uma maneira de compreender como estes ocorrem. Isso pode abrir espaço para as empresas resolverem problemas como cronogramas atrasados de forma ágil. Por fim, é possível melhorar a eficiência e confiabilidade do serviço oferecido à população e talvez contribuir sustentavelmente para a cidade fazendo mais pessoas optarem pelo transporte público.

Nesse contexto, foram definidas métricas baseadas em parâmetros como *headway* e distância, que são usadas na implementação de algoritmos para detecção de aglomerados de ônibus em tempo real para uma aplicação de monitoramento de ônibus chamada OndeBUS. O foco deste trabalho é avaliar o desempenho dos algoritmos e comparar seus resultados, sendo assim possível ter uma ideia de qual algoritmo tem maior eficácia na detecção de aglomerados.

2 FUNDAMENTAÇÃO TEÓRICA

No contexto de transporte público, especificamente de ônibus, alguns termos são especialmente importantes.

Um **trajeto** ou **rota** é um grupo de viagens exibidas aos passageiros como um único serviço. Uma **viagem** é uma sequência de duas ou mais paradas executadas por um ônibus que ocorrem em um horário específico. Os **horários de parada** são os respectivos horários de partida e chegada dos veículos em paradas específicas em cada viagem. Outro termo importante é o **headway (h)** que corresponde ao intervalo de tempo (em minutos) entre os horários de dois ônibus de uma mesma rota, medidos em uma determinada parada de ônibus. Por fim, há os **shapes** que são os trajetos programados que os ônibus devem executar, compostos por pontos de geolocalização.

A Especificação Geral de Feeds de Transporte Público (GTFS¹) define um formato comum para horários de transportes públicos e informações geográficas associadas. Os feeds GTFS permitem que as agências de transporte público publiquem suas informações e que os desenvolvedores criem aplicativos que consumam esses dados com interoperabilidade.

¹ GTFS - Especificação Geral de Feeds de Transporte Público <<https://developers.google.com/transit/gtfs>>

2.1 Trabalhos Relacionados

Os trabalhos relacionados foram categorizados em duas partes: aplicações de monitoramento de ônibus e métricas de definição de aglomerados de ônibus.

2.1.1 Aplicações de Monitoramento de Ônibus

No contexto de monitoramento de ônibus em tempo real, diversas aplicações já foram desenvolvidas, as quais buscam, em geral, fornecer informações sobre ônibus em tempo real. Um exemplo é o CIOM-CG² da STTP de Campina Grande, que exibe os ônibus executando suas rotas. Já o Google Maps³ da Google e o site Olho Vivo⁴ da SPTrans de São Paulo, exibem a localização em tempo real de uma linha, informações de paradas e os horários que os ônibus irão chegar nelas. Nos aplicativos Cadê o Ônibus?⁵ e Vá de Ônibus⁶, é possível calcular rotas, localizar os pontos mais próximos, consultar itinerários e acompanhar a localização dos ônibus em tempo real de São Paulo e Rio de Janeiro, respectivamente. Ainda, o Moovit⁷ é um aplicativo que dispõe de informações sobre linhas de ônibus, metrô, trem e até barcas e teleféricos, mostra os horários e os itinerários das linhas e permite comparar trajetos com informações sobre horários em tempo real. Basicamente, essas aplicações mostram, de variadas maneiras, os ônibus realizando suas rotas no mapa em tempo real, enquanto nenhuma aplicação foi encontrada capaz de detectar aglomerados de ônibus em tempo real.

2.1.2 Detecção de Aglomerados de Ônibus

Em relação à ocorrência de aglomerados de ônibus, diversos autores já investigaram o impacto de fatores nos níveis de serviço. FENG & FIGLIOZZI (2011) apontaram que aglomerados de ônibus, definidos por um *headway* de menos de 3 minutos para a TriMet Route 15 em Portland, aumentaram de acordo com o comprimento da rota. Os autores usaram dados ao nível de parada, coletados de tecnologias como Contagem Automática de Passageiros e Localização Automática de Veículos (em inglês, *Automatic Vehicle Location* - AVL and *Automatic Passenger Count* - APC, respectivamente) e visualizaram os aglomerados plotando os registros de *headway* observados. Os

² <https://www.ciomcg.com.br/>

³ <https://www.google.com.br/maps>

⁴ <http://olhovivo.sptrans.com.br/>

⁵ <https://apps.apple.com/br/app/cad-c3-aa-o-c3-b4nibus/id815994264>

⁶ <https://www.vadeonibus.com.br>

⁷ <https://moovitapp.com/>

aglomerados aumentaram ao longo do comprimento da rota, mas em um certo ponto de tempo o atraso diminuiu, sugerindo que esses pontos temporais ajudam a manter *headways* regulares.

O modelo baseado em regressão linear usando o atraso de *headway* (em segundos) como variável dependente, proposto por EL-GENEIDY et al. (2016), mostra que uma das principais causas dos aglomerados de ônibus é o atraso no início da rota para o primeiro ônibus. O modelo indica que, se o primeiro ônibus começar a executar a rota atrasado, isso aumentará as chances de ocorrer um aglomerado com o ônibus seguinte mais do que seu atraso em outro ponto da rota. Portanto, eles concluíram que, para prevenir a ocorrência de aglomerados, o foco deve ser rastrear atrasos no início das rotas, uma vez que diminuiria as chances de serem aglomerados na próxima viagem programada, quando evitada.

MUNIZAGA et al. (2018) constataram que os aglomerados de ônibus são entendidos como a instabilidade de *headways*. Eles classificam a detecção de aglomerados por *headway* como indicadores discretos. Esses indicadores baseiam-se na observação do fenômeno através de uma variável que pode assumir valores específicos, ou seja, quando o *headway* observado é menor do que uma porcentagem do *headway* programado ou quando o *headway* observado está dentro de um determinado intervalo, com base no *headway* programado ou na hora de chegada programada de um ônibus em determinada parada.

Esses trabalhos realçam a importância de compreender o ambiente volátil em que os ônibus se encontram e seu impacto na definição de limiares para detecção de aglomerados que sejam apropriados e coerentes para a detecção.

3 SOLUÇÃO

Esta seção tem como objetivo introduzir a arquitetura da aplicação com as modificações propostas, definir as métricas e mostrar o processo de implementação dos algoritmos de detecção de aglomerados de ônibus na aplicação e exibí-la em uso.

3.1 Arquitetura da Aplicação

O OndeBUS é uma aplicação para auxiliar o monitoramento de frotas de ônibus das cidades. A aplicação foi inicialmente desenvolvida pelos alunos de pós-graduação do Laboratório de Qualidade de Dados (LQD) da UFCG. A aplicação exibe em tempo real em um

mapa os ônibus de várias linhas executando seus trajetos ao longo do dia.

Os dados usados na aplicação são pré-processados a partir de dados de GPS⁸ enviados pelos ônibus, que contém informações de geolocalização, horário que o sinal de GPS foi enviado, código e rota em tempo real, e GTFS, que fornece informações estáticas sobre sequências e pontos dos shapes, paradas, comprimento das rotas de ida, volta, e circular, dados esses que são fornecidos pelas próprias empresas de ônibus.

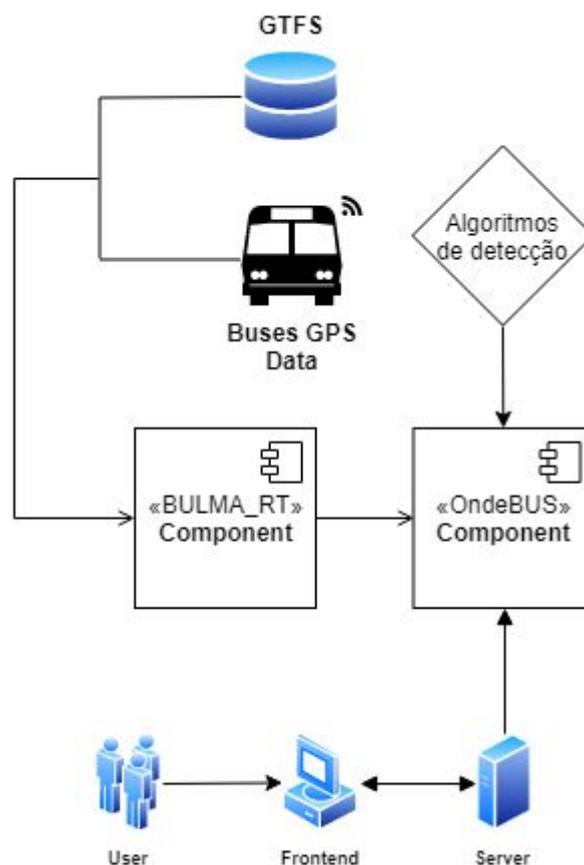


Figura 1: Diagrama de componentes/fluxo do OndeBUS.

O OndeBUS é uma aplicação Web com *back-end* em Java, que também faz uso de JPA (*Java Persistence API*), uma API padrão da linguagem Java que descreve uma interface comum para *frameworks* de persistência de dados, e Apache Tomcat 8.0, um servidor web Java, mais especificamente, um *container de servlets*.

A aplicação conta com *front-end* em JSF (*JavaServer Faces*), uma tecnologia que permite criar aplicações Java para Web utilizando componentes visuais pré-prontos, usada

⁸ GPS - Global Positioning System

em conjunto com o *PrimeFaces*, um *framework* para projetos JSF.

Na execução do OndeBUS faz-se uso do BULMA Real Time (MESTRE, 2018), uma biblioteca .jar responsável por fazer o pré-processamento dos dados de GPS e GTFS. Uma outra biblioteca .jar chamada *StreamSimulationFile* é responsável por simular o envio em tempo real desses dados para a aplicação, como se fossem os próprios ônibus enviando seus sinais de GPS, a fim de facilitar os testes de execução.

De acordo com o fluxo de execução da aplicação (ver Figura 1), os ônibus enviam dados de GPS, que são pré-processados pelo componente BULMA Real Time, associando os dados de GPS com os dados do GTFS de forma a se obter dados padronizados para uso na aplicação. Em seguida, os dados padronizados são coletados pelo *back-end* (na figura denotada pelo componente ‘OndeBUS’) onde é criada uma lista dos ônibus em movimento no horário atual, contendo os pontos de GPS com informações que serão exibidas na visualização. O componente ‘OndeBUS’ é o responsável por gerar informações de monitoramento e será nele onde serão implementados os algoritmos para a detecção dos aglomerado de ônibus.

Por fim, as informações são enviadas para o servidor, processadas pela interface e consumidas pelo usuário.

3.2 Implementação

Nesta seção, são mostrados os pseudocódigos de detecção de aglomerados de ônibus, assim como a aplicação original e modificada do OndeBUS. Os dados reais utilizados são das linhas de ônibus da cidade de Campina Grande, Paraíba, ao longo do dia 21 de outubro de 2017, simulados em tempo real para testar a execução da aplicação.

3.2.1 *Back-end*

Nesse ponto, são aplicadas as métricas descritas nas Seções 2.2.1 e 2.2.2. Na detecção por distância (ver Figura 3), os ônibus em movimento são agrupados por rota. Em seguida, a distância (em metros) de um ônibus em relação aos outros ônibus da mesma rota é obtida juntamente com o tamanho e o número de ônibus da rota. Com isso, calcula-se o limiar e define-se a ocorrência de aglomerados. Na detecção por *headway* (ver Figura 2), são utilizadas as paradas de ônibus, organizadas por rota, como um dos parâmetros para a detecção de aglomerados, onde os mesmos contém um atributo indicando o ponto de *shape*

mais próximo e um dado de GPS de um ônibus contém o ponto de *shape* em que o mesmo se encontra. Em seguida, procura-se paradas entre o último e o penúltimo ponto de GPS que o ônibus passou, e calcula-se, por meio de interpolação, o horário que o ônibus passou na parada, assim como o *headway* programado, agrupando essas informações por parada. Quando um segundo ônibus passa pela mesma parada do primeiro ônibus, calcula-se o *headway* observado entre eles e compara-se com o limiar já caracterizado como vinte e cinco por cento do *headway* programado.

Algoritmo *headway*

1. recebe dados de GPS dos ônibus em movimento, lista de paradas de ônibus mapeadas por rota e sequência da paradas, lista de *headways* dos ônibus mapeados por rota e parada
2. **para** cada onibus *o* em onibusEmMovimento OM
3. rota *r* <- *o*.ultimoSinalGPS.rota
4. sequenciaParada <- 1
5. **enquanto** todasAsParadas.*r* contiverem sequenciaParada
6. **para** parada *p* em todasAsParadas.rota.sequenciaParada
7. **se** *p* estiver entre *o*.ultimoSinalGPS e *o*.penultimoSinalGPS
8. headwayProgramado <- *p*.arrivalTime - (*p*+1).arrivalTime
9. horarioOnibusNaParada <- interpolacao(*o*, *p*)
10. adiciona informação de headway do ônibus em listaHeadways
11. **se** listaHeadways.rota.parada.tamanho > 1
12. onibus1 <- listaHeadways.rota.parada(1)
13. onibus2 <- listaHeadways.rota.parada(2)
14. headwayReal <- onibus1.horarioOnibusNaParada - onibus2.horarioOnibusNaParada
15. **se** headwayReal < (headwayProgramado / 4)
16. adiciona os ônibus na lista de aglomerados
17. **fim-se**
18. **fim-se**
19. **fim-se**
20. sequenciaParada += 1
21. **fim-para**
22. **fim-enquanto**
23. **fim-para**
24. retorna lista com código e rota de ônibus aglomerados

Figura 2: Pseudocódigo para detecção de aglomerados de ônibus por *headway*.

Algoritmo **distância**

1. recebe dados de GPS dos ônibus em movimento
2. agrupa os ônibus em movimento por rota em uma lista
3. **para** cada rota
4. onibusEmRota <- lista de ônibus naquela rota
5. limiar <- (tamanho da rota / #onibusEmRota) / 4
6. **enquanto** (#onibusEmRota > 1)
7. onibus1 <- retorna primeiro elemento de onibusEmRota e remove
8. **para** cada onibus2 em onibusEmRota
9. distancia <- distanciaEntre(onibus1,onibus2)
10. tempo <- tempoEntre(onibus1,onibus2)
11. **se** (distancia < limiar e tempo < 60)
12. adiciona os ônibus na lista de aglomerados
13. **fim-se**
14. **fim-para**
15. **fim-enquanto**
16. **fim-para**
17. retorna lista com código e rota de ônibus aglomerados

Figura 3: Pseudocódigo para detecção de aglomerados de ônibus por distância.

3.2.2 *Front-end*

A aplicação OndeBUS originalmente, além de exibir os ônibus executando suas rotas em tempo real em um mapa, como mostrado na Figura 4, fornece uma representação visual dos ônibus classificados por cores que indicam quais estão regulares, atrasados, adiantados e fora da rota. Os ícones dos ônibus também são rotulados por código. É possível filtrar os ônibus por rota e código.

Depois da implementação dos algoritmos e modificações na interface gráfica, para exibir os ônibus aglomerados a partir da detecção de aglomerados proposta, foi considerada uma representação em cores que pudesse diferenciar os ônibus aglomerados dos que não estão, como visto na Figura 5 e Figura 6. Foram adicionadas duas novas cores de ônibus: a roxa, que indica que um ônibus está aglomerado, e a cinza para um ônibus não aglomerado. Ao marcar na tela a opção ‘Show Bus Bunching’ os ônibus aglomerados, apenas as duas cores citadas predominam no mapa, a fim de evitar que as outras cores não se sobreponham, tornando mais limpa e clara a visualização dos aglomerados.

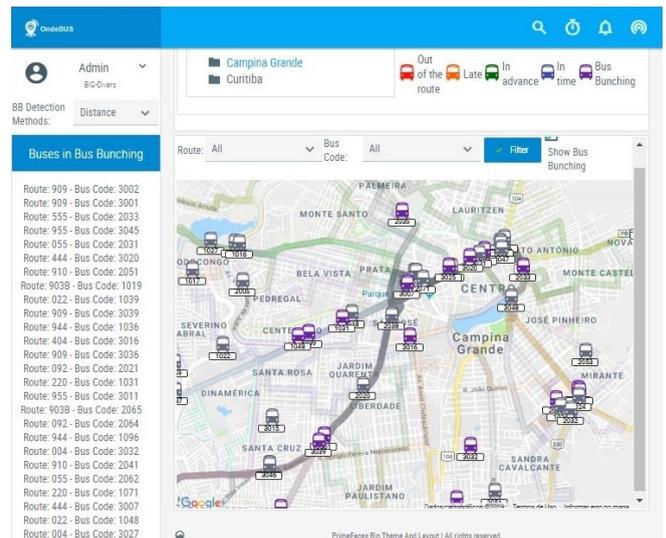


Figura 5: Detecção de aglomerados de ônibus por distância. (Versão Modificada)

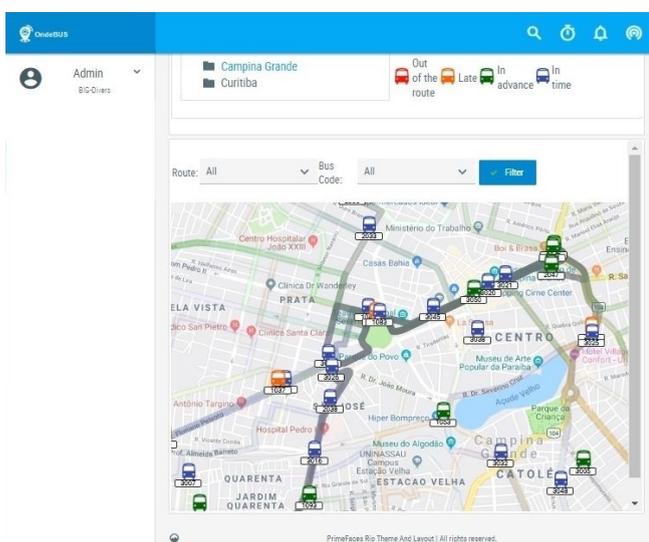


Figura 4: Monitoramento dos ônibus em tempo real pelo OndeBUS. (Versão Original)

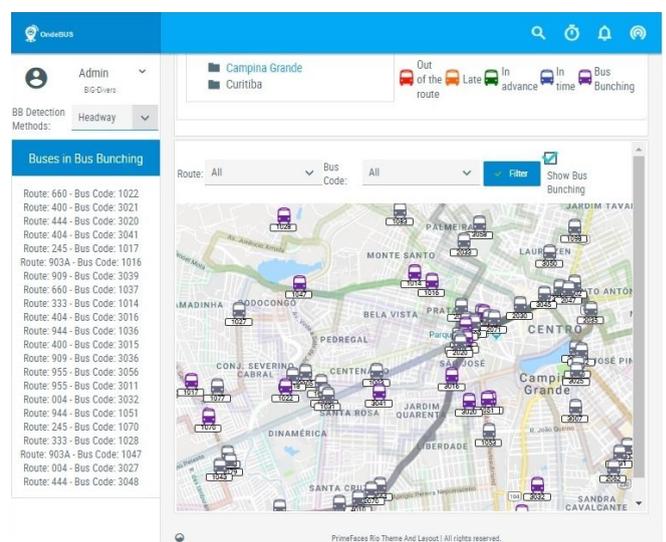


Figura 6: Detecção de aglomerados de ônibus por headway. (Versão Modificada)

Pode-se também mudar, em tempo de execução, o tipo de detecção de aglomerados. Nesse caso, o usuário pode escolher o método de detecção de aglomerados (distância ou *headway*) desejado. Além disso, há também outra forma de visualizar os resultados da detecção de aglomerados: um painel que exibe informações dos ônibus aglomerados que não estão no mapa, tais como código do ônibus associados a qual rota o mesmo pertence, facilitando a identificação dos pares de ônibus (ver Figura 7). O usuário também pode aplicar filtros para visualizar apenas os ônibus aglomerados de uma determinada rota.

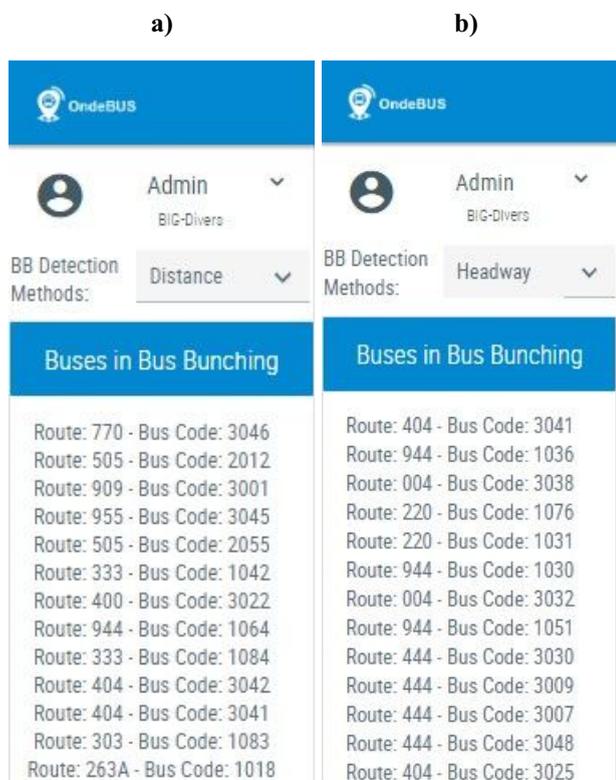


Figura 7: Filtro de detecção de aglomerados: a) por distância e b) por *headway*.

4 AVALIAÇÃO DOS ALGORITMOS

Nessa seção, serão denotadas a metodologia utilizada para definição das métricas e limiares utilizados nos algoritmos de detecção de aglomerados de ônibus e discutidos os resultados obtidos pela a aplicação após a implementação dos algoritmos usando as métricas propostas.

4.1 Métricas para Detecção de Aglomerados de Ônibus

A detecção de aglomerados de ônibus é possível a partir da utilização de métricas derivadas do *headway* e da distância entre um par de ônibus executando a mesma rota.

Utilizando o *headway*, a ocorrência de aglomerados de ônibus pode ser medida pela regularidade do *headway* e deve ser definida com base em um limiar. O Manual de Capacidade de Trânsito e Qualidade de Serviço (em inglês, *Transit Capacity and Quality of Service Manual - TCQSM*) recomenda usar a probabilidade de que o *headway* real de um veículo de transporte público seja zero ou menor que a metade do *headway* programado para identificar irregularidades de *headway* (Haiyang Yu et al., 2016).

Para a métrica de distância, segue-se a mesma linha de pensamento da métrica anterior. Neste caso, a ocorrência de aglomerados é medida usando a distância entre dois pontos de GPS enviados por um par de ônibus, e pela diferença entre os horários que os sinais foram enviados, como um tempo limite para erro de atraso no envio do sinal.

4.1.1 Detecção de Aglomerados por *Headway*

Na prática, pode-se ver na Figura 8 que, no evento (a) os ônibus 'c' e 'd' mantêm um *headway* estável entre si, seguindo seu caminho sem aglomerar. Logo em seguida, no evento (b), onde em uma das paradas há vários passageiros, o tempo de 'c' na parada aumenta, ocasionando o atraso na saída de 'c' da parada, enquanto 'd' segue se aproximando, estreitando o *headway* entre eles. Por último, na situação (c) é quando os ônibus 'c' e 'd' finalmente se encontram na parada zerando o *headway* entre eles caracterizando a ocorrência de um aglomerado entre o par de ônibus.

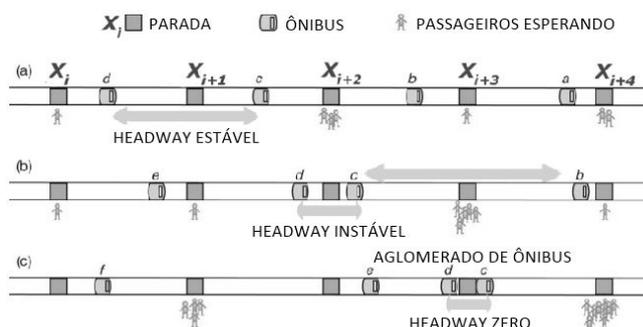


Figura 8: Simulação de um aglomerado por *headway*, adaptado de [3].

De acordo com as definições de MUNIZAGA et al., o aglomerado acontece quando:

$$h_{\text{observado}} < \frac{h_{\text{programado}}}{4} \quad (1)$$

ou seja, quando o *headway* observado, também conhecido como *headway* real, for menor que vinte e cinco por cento do *headway* programado naquela parada de ônibus; caso contrário, os dois ônibus não estão executando a rota juntos.

A divisão por quatro foi sugerida pelos autores MUNIZAGA et al. para aumentar o intervalo de confiança da detecção para situações mais críticas onde os ônibus não estão mais seguindo seus cronogramas e evitar casos falsos positivos. Por exemplo, caso o *headway* observado fosse um segundo menor que o programado, não necessariamente caracterizaria um aglomerado, porém de acordo com a Equação 1, se o *headway* observado estiver abaixo de vinte e cinco por cento do valor programado, isso indica um estreitamento do *headway* real dos ônibus.

4.1.2 Detecção de Aglomerados por Distância

Não foram encontrados trabalhos que usem a distância como parâmetro para a detecção de aglomerados em tempo real. Logo, para encontrar um limiar foram considerados de forma exploratória fatores como: rotas de alta e baixa frequência, distribuição uniforme dos ônibus, tamanho da rota e número de ônibus que a executam. Assim, para manter um limiar ajustado para cada rota, chegou-se a um entendimento considerando o impacto desses fatores no desempenho de cada rota, onde em um ambiente estocástico, caracterizado pela Figura 9, todos os ônibus de uma mesma rota deveriam estar a uma mesma distância um do outro. Caso algum par de ônibus desobedeça essa distância uniforme, tal par iria caracterizar um aglomerado.

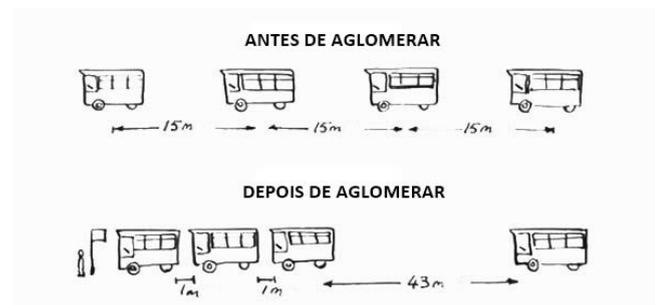


Figura 9: Simulação de um aglomerado por distância, adaptado de [8].

Considerando a distância entre dois ônibus A e B, tem-se que, se esse valor for menor que vinte e cinco por cento do tamanho da rota dividido pela quantidade de ônibus que a executam simultaneamente, acontece o aglomerado entre o par de ônibus, definido pela condição:

$$\text{distância}_{AB} < \frac{\text{tamanho da rota} / \text{n}^\circ \text{ de ônibus}}{4} \quad (2)$$

Percebe-se que foi usada a mesma divisão por quatro do método anterior na tentativa obter um intervalo de confiança que valide melhor os resultados da métrica na detecção de aglomerados como na seção anterior.

Um limiar de tempo entre o horário de envio dos dados de cada par de ônibus também foi utilizado. Considerando que a detecção ocorre em tempo real, não seria lógico termos os sinais de GPS enviados por um par de ônibus com um intervalo de tempo muito longo entre eles. Se isso acontece, a distância entre os ônibus não seria precisa quanto à localização real em que se encontram. Diante desse problema, foi usado um limiar arbitrário de tempo de sessenta segundos entre os pares de ônibus.

4.2 Discussão

Foram coletados dados das detecções de aglomerados de ônibus, simulados em tempo real no horário entre 11h e 14h (horário de pico) do dia 21 de outubro de 2017. Os dados foram exportados pela aplicação para arquivos externos no formato CSV, que eram populados dinamicamente à medida que a detecção dos aglomerados ocorria, contando com diversas informações sobre os ônibus aglomerados. O estudo terá caráter essencialmente qualitativo, já que será necessário o cruzamento dos dados levantados com toda a pesquisa bibliográfica já feita.

Na Figura 10, é possível ver no histograma de detecção por *headway*, que mais de 75% dos pares de ônibus aglomerados apresentam *headways* observados no intervalo de 0-75 segundos, e para *headways* acima de 125 segundos, ocorrem poucas detecções. Esses resultados confirmam a detecção eficaz dos aglomerados de ônibus, pois, como citado na Seção 4.1, este evento acontece quando o *headway* está próximo de zero.

Já na detecção por distância, os resultados (Figura 11) apresentam-se mais dispersos no histograma, porém com concentração também à esquerda, onde as distâncias entre os pares de ônibus são menores. Pode-se visualizar que grande parte das detecções ocorreram quando os pares de

ônibus estavam a um intervalo de 750-2.000 metros de distância.

Apesar de se encontrarem dentro do limiar de distância estabelecido (Equação 2), os resultados mais à direita do gráfico da Figura 11 não fornecem garantia de que os pares de ônibus realmente estejam aglomerados, uma vez que, na definição teórica, um aglomerado de ônibus é caracterizado

quando dois ônibus executam a mesma rota juntos (FENG & FIGLIOZZI, 2011). Isto acontece, em geral, com rotas de grande extensão e poucos ônibus servindo-as, porque, quanto maior o tamanho da rota, maior o espaçamento entre os ônibus, refletindo diretamente no aumento do limiar.

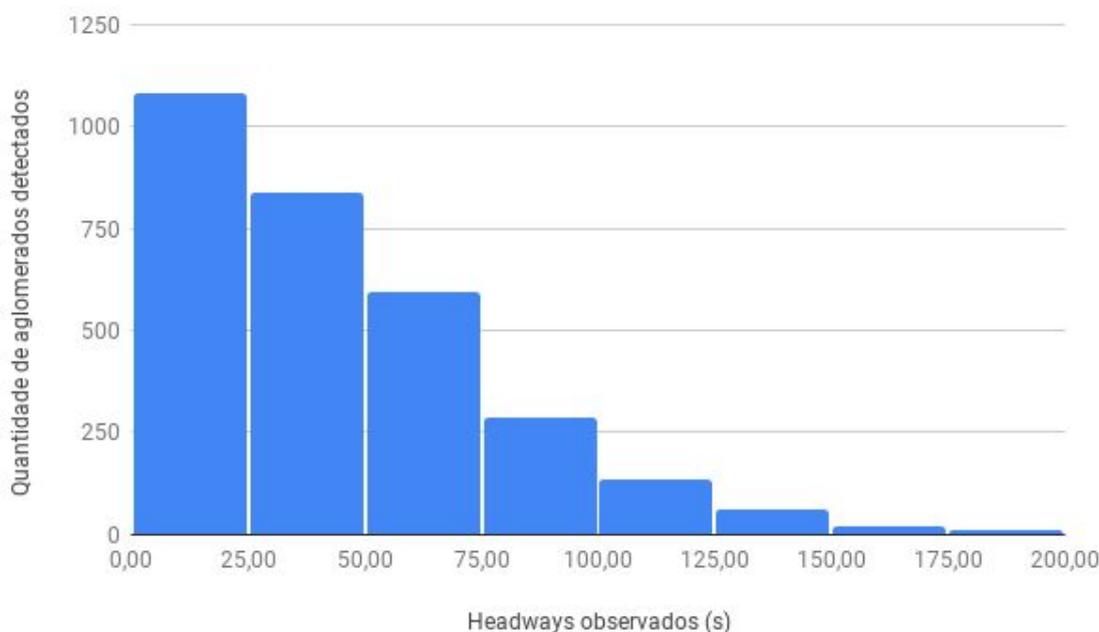


Figura 10: Histograma com resultados do algoritmo de detecção por *headway*.

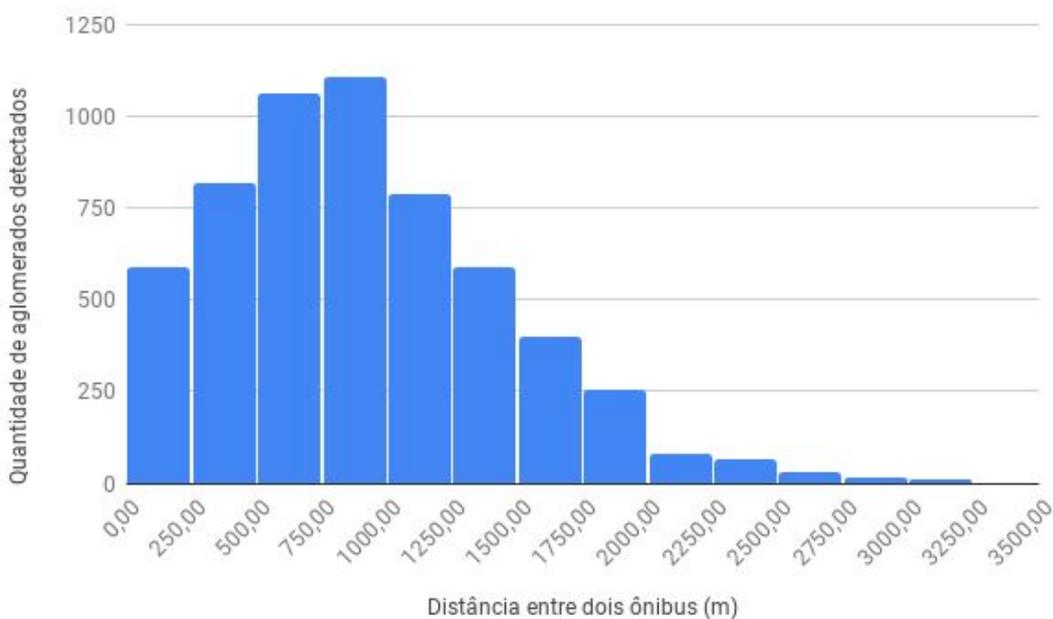


Figura 11: Histograma com resultados do algoritmo de detecção por distância.

5 CONCLUSÃO

Nesse projeto, foram usados dados de GPS e GTFS dos ônibus simulados em tempo real para desenvolver algoritmos capazes de detectar aglomerados de ônibus, utilizando métricas baseadas em distância e *headway*.

Foi possível identificar visualmente a ocorrência de aglomerados de ônibus na ferramenta, devido à escolha de cores contrastantes para exibir a ocorrência dos aglomerados. Além disso, a ferramenta conta com o painel lateral de informações adicionais sobre os ônibus que compõem o evento dos aglomerados.

Utilizando o algoritmo de *headway*, foi possível identificar aglomerados de forma eficaz, em geral de 0 a 75 segundos, de acordo com os dados analisados.

Utilizando o algoritmo de distância, também foi possível identificar aglomerados de forma eficaz, em geral de 0 a 600 metros, de acordo com os dados analisados. Porém, foram detectados alguns ônibus aglomerados com distância de até 3.000 metros, o que contradiz a definição de aglomerado de ônibus de FENG & FIGLIOZZI.

Comparando os dois algoritmos, a detecção por *headway* apresenta melhores resultados do que a métrica de distância. Inúmeros trabalhos utilizando de *headway* facilitam bastante na definição do limiar e dos cálculos por já por ser um parâmetro bastante utilizado no meio de detecção de aglomerados. Por contradizer a definição de aglomerados em alguns casos e na prática não ter-se conhecimento de uma distância base entre os ônibus que possam caracterizar a ocorrência de aglomerados ou algum gabarito fornecido pelas empresas de ônibus que possam validar os dados, os resultados do algoritmo de detecção de distância se tornam mais duvidosos.

5.1 Processo de Desenvolvimento

O processo de desenvolvimento dos algoritmos se deu inicialmente pela necessidade de uma revisão literária para compreender o problema dos aglomerados de ônibus. O foco principal da maioria dos artigos era na área de análise de dados temporais dos ônibus, considerando possíveis causas dos aglomerados e como os mesmos impactam diretamente na ocorrência, vendo onde e em que horário do dia ocorre o picos de ocorrência de aglomerados, com o intuito de encontrar uma maneira de prevenir a ocorrência futura dos aglomerados naqueles lugares.

Em seguida, uma pesquisa para definir limiares de distância e *headway* foi realizada. As métricas são parte

crucial no processo de desenvolvimento, pois elas são a validação da solução, todo o trabalho foi feito em cima dessas métricas que forneceram dados confiáveis para a análise dos dados de aglomerados na aplicação.

O processo de implementação seguiu o modelo de desenvolvimento de software ágil baseado em Scrum, definido por sprints semanais, onde na entrega sempre era definido o *backlog* (conjunto de requisitos) da próxima sprint.

5.2 Limitações

Uma das limitações principais foi a falta de trabalhos relacionados com a métrica de distância, não houve muita base teórica que pudesse ser utilizada para a definição da mesma.

Testar a ferramenta com dados reais de outras cidades também foi planejado, porém não foi possível realizar esse teste por questões de prazo para com a entrega.

5.3 Trabalhos Futuros

O próximo passo a ser dado nesse projeto poderia ser estudar a definição da métrica de distância, com intuito de deixá-la mais adequada à definição de FENG & FIGLIOZZI de aglomerados e minimizar o aparecimento desses possíveis falsos positivos.

Também em busca de melhorar a precisão dos resultados da aplicação, seria interessante a implementação de um método de detecção híbrido, que fizesse uso das métricas de distância e *headway* simultaneamente.

Testar a aplicação com dados de outros dias e cidades para analisar a detecção das métricas e descobrir se as mesmas se comportam de maneira distinta da vista neste trabalho.

REFERÊNCIAS

- [1] DIAB, Ehab; BERTINI, Robert; EL-GENEIDY, Ahmed. Bus transit service reliability: Understanding the impacts of overlapping bus service on headway delays and determinants of bus bunching. In: 95th Annual Meeting of the Transportation Research Board. 2016.
- [2] FENG, W.; FIGLIOZZI, M. Using archived AVL/APC bus data to identify spatial-temporal causes of bus bunching. Compendium of papers of 90th

- Transportation Research Board. In: Annual Meeting, Washington, DC. 2011.
- [3] MOREIRA-MATIAS, Luis et al. Bus bunching detection: A sequence mining approach. In: Workshop on Ubiquitous Data Mining. 2012. p. 13.
- [4] MUNIZAGA, Marcela; ARRIAGADA, Jacqueline; GSCHWENDER, Antonio. Modeling Bus Bunching Using Massive Location and Fare Collection Data. Journal of Intelligent Transportation Systems, p. 1-13, 2018.
- [5] YU, Haiyang et al. Headway-based bus bunching prediction using transit smart card data. Transportation Research Part C: Emerging Technologies, v. 72, p. 45-59, 2016.
- [6] SITE MUNDO EDUCAÇÃO. A qualidade do transporte público no Brasil e os protestos. Disponível em: <<https://mundoeducacao.bol.uol.com.br/geografia/a-qualidade-transporte-publico-no-brasil-os-protestos.htm>> Acesso em: 08 de dezembro de 2018.
- [7] KITTELSON & ASSOCIATES; TRANSIT COOPERATIVE RESEARCH PROGRAM; TRANSIT DEVELOPMENT CORPORATION. Transit capacity and quality of service manual. Transportation Research Board, 2003.
- [8] SITE GUERNSEYDONKEY.COM. Why do Buses Come in Threes ?. Disponível em: <<https://guernseydonkey.com/why-do-buses-come-in-threes/>> Acesso em: 18 de junho de 2019.
- [9] MESTRE, D. G. Leveraging the entity matching performance through adaptive indexing and efficient parallelization. 2018. 155 f. (Tese de Doutorado em Ciência da Computação) Programa de Pós-graduação em Ciência da Computação, Centro de Engenharia Elétrica e Informática, Universidade Federal de Campina Grande - Paraíba - Brasil, 2018.