

PGCOMP - Programa de Pós-Graduação em Ciência da Computação  
Universidade Federal da Bahia (UFBA)  
Av. Milton Santos, s/n - Ondina  
Salvador, BA, Brasil, 40170-110

<https://pgcomp.ufba.br>  
[pgcomp@ufba.br](mailto:pgcomp@ufba.br)

Os Sistemas de Transporte Inteligente (ITS) são projetados para analisar o fluxo de veículos nas rodovias com o objetivo de identificar eventuais anomalias de tráfego, garantindo maior eficiência na tomada de decisão. Estes sistemas podem ser estruturados no contexto das redes veiculares Ad-Hoc (VANET), nas quais realizam a tarefa de integrar os elementos do espaço urbano por meio de um sistema de comunicação distribuído. Analogamente, aplicações ITS requerem um monitoramento constante das rodovias e tal monitoramento visa analisar, entre outros aspectos, a variação da densidade de veículos ao longo do tempo. Em geral, este processo ocorre por meio do envio periódico de dados situacionais do ambiente de mobilidade para a nuvem. Como consequência, os conjuntos de dados enviados em alta frequência para nuvem formam um fluxo contínuo de dados a serem processados e essa dinâmica resulta no aumento progressivo do custo de comunicação. Esse fenômeno ocorre em função do volume de dados transferidos no enlace entre a névoa e a nuvem, elevando os riscos de sobrecarga no canal de comunicação, além de aumentar a latência durante as requisições de serviços disponibilizados na nuvem. Diante disso, propõe-se neste trabalho, a elaboração de uma arquitetura multicamada para redução de dados baseada na computação em névoa (Fog Computing) denominada DaRkaM, acrônimo em inglês para (Data Reduction Framework for Traffic Management). O método consiste na utilização de uma estratégia de monitoramento capaz de realizar processos de redução de dados diretamente na borda da rede veicular. Na camada da nuvem, DaRkaM atua como um controlador central, analisando as posições geográficas dos veículos que são recebidas a partir de um fluxo de dados contínuo, os quais são utilizados nos processos de monitoramento e gestão do tráfego. Na borda da rede, um módulo de redução de dados implementado no framework DaRkaM foi projetado para hospedar diferentes estratégias de monitoramento de tráfego. Esta arquitetura permite a realização de análises comparativas envolvendo tanto a utilização de algoritmos baseados em amostragem simples, como o uso de algoritmos de agrupamento, nos quais os processos de redução de dados são estruturados com base na similaridade dos objetos analisados. Os resultados demonstraram que a utilização de algoritmos baseados em agrupamento, hospedados no núcleo de redução de dados do framework DaRkaM, alcançaram uma significativa redução no custo de comunicação, especialmente em cenários sobrecarregados, além de manterem alta precisão no monitoramento e detecção de congestionamento de tráfego.

Palavras-chave: Redução de Dados, Agrupamentos, Aprendizado de Máquina, Computação em Névoa, Redes Ad Hoc Veiculares, Sistemas de Transporte Inteligente.

# DaRkaM - Um Framework de Redução de Dados Baseado na Névoa Aplicado ao Contexto da Computação Urbana

Edson Mota da Cruz

Tese de Doutorado

Universidade Federal da Bahia

Programa de Pós-Graduação em  
Ciência da Computação

Fevereiro | 2023

DSC | 038 | 2023

DaRkaM: Um Framework de Redução de Dados Baseado na Névoa Aplicado ao Contexto da Computação Urbana

Edson Mota da Cruz

UFBA





Universidade Federal da Bahia  
Instituto de Computação

Programa de Pós-Graduação em Ciência da Computação

**DARKAM: UM FRAMEWORK DE REDUÇÃO  
DE DADOS BASEADO NA NÉVOA  
APLICADO AO CONTEXTO DA  
COMPUTAÇÃO URBANA**

Edson Mota da Cruz

TESE DE DOUTORADO

Salvador  
01 de fevereiro de 2023



EDSON MOTA DA CRUZ

**DARKAM: UM FRAMEWORK DE REDUÇÃO DE DADOS  
BASEADO NA NÉVOA APLICADO AO CONTEXTO DA  
COMPUTAÇÃO URBANA**

Esta Tese de Doutorado foi apresentada ao Programa de Pós-Graduação em Ciência da Computação da Universidade Federal da Bahia, como requisito parcial para obtenção do grau de Doutor em Ciência da Computação.

Orientador: Professor Dr. Maycon Leone Maciel Peixoto

Salvador

01 de fevereiro de 2023

Ficha catalográfica elaborada pela Biblioteca Universitária de  
Ciências e Tecnologias Prof. Omar Catunda, SIBI – UFBA.

C957 Cruz, Edson Mota da

DaRkaM: Um Framework de Redução de Dados Baseado  
na Névoa Aplicado ao Contexto da Computação Urbana/  
Edson Mota da Cruz. – Salvador, 2023.

157 f.

Orientador: Prof. Dr. Maycon Leone Maciel Peixoto

Tese (Doutorado) – Universidade Federal da Bahia, Instituto  
de Computação, 2023.

1. Computação. 2. Algoritmos. 3. Arquitetura –  
computadores. 4. Sistemas de transporte inteligente de veículo.  
I. Peixoto, Maycon Leone Maciel. II. Universidade Federal da  
Bahia. III Título.

CDU 004.2

**EDSON MOTA DA CRUZ**

**DARKAM: UM FRAMEWORK DE REDUÇÃO DE DADOS BASEADO  
NA NÉVOA APLICADO AO CONTEXTO DA COMPUTAÇÃO  
URBANA**

Esta tese foi julgada adequada à obtenção do título de Doutor em Ciência da Computação e aprovada em sua forma final pelo Programa de Pós-Graduação em Ciência da Computação da UFBA.

Salvador, 01 de fevereiro de 2023



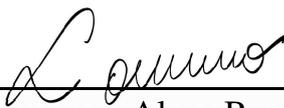
Prof. Dr. Maycon Leone Maciel Peixoto (Orientador -UFBA)



Prof. Dr. Geraldo Pereira Rocha Filho (UESB)



Prof. Dr. Dionisio Machado Leite Filho (UFMS)



Prof. Dr. Lourenço Alves Pereira Júnior (ITA)



Prof. Dr. Ademair Takeo Akabane (PUC-Campinas)



*Dedico este trabalho à minha esposa Aline Câmara, pela sua paciência e confiança, sem as quais este trabalho não teria o mesmo desfecho, aos meus pares de pesquisa e professores que tanto contribuíram para o desenvolvimento deste trabalho.*



## **AGRADECIMENTOS**

Registro aqui meus sinceros agradecimentos aos meus familiares, amigos pesquisadores e profissionais que contribuíram com discussões, ideias, apoio técnico sobre diversos aspectos deste projeto.

Agradeço aos professores da UFBA pela competência e qualidade técnica, além da disponibilidade para mostrar os caminhos que levam ao conhecimento.

Entre estes, agradeço em especial ao Prof. Dr. Maycon Leone, pela sua contribuição na definição dos objetivos, ajuda, compreensão, paciência e, principalmente, sua capacidade de inspirar os seus alunos na busca pelos melhores resultados, fazendo isso com empatia, humildade e dedicação.

Agradeço a Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES) pelo apoio financeiro que viabilizou a realização desta pesquisa.

E a todos que participaram direta ou indiretamente deste trabalho.



*Somos o que pensamos. Tudo o que somos surge com nossos pensamentos. Com nossos pensamentos fazemos o nosso mundo.*

— (Buda)



## RESUMO

Os Sistemas de Transporte Inteligente (ITS) são projetados para analisar o fluxo de veículos nas rodovias com o objetivo de identificar eventuais anomalias de tráfego, garantindo maior eficiência na tomada de decisão. Estes sistemas podem ser estruturados no contexto das redes veiculares Ad-Hoc (VANET), nas quais realizam a tarefa de integrar os elementos do espaço urbano por meio de um sistema de comunicação distribuído. Analogamente, aplicações ITS requerem um monitoramento constante das rodovias e tal monitoramento visa analisar, entre outros aspectos, a variação da densidade de veículos ao longo do tempo. Em geral, este processo ocorre por meio do envio periódico de dados situacionais do ambiente de mobilidade para a nuvem. Como consequência, os conjuntos de dados enviados em alta frequência para nuvem formam um fluxo contínuo de dados a serem processados e essa dinâmica resulta no aumento progressivo do custo de comunicação. Esse fenômeno ocorre em função do volume de dados transferidos no enlace entre a névoa e a nuvem, elevando os riscos de sobrecarga no canal de comunicação, além de aumentar a latência durante as requisições de serviços disponibilizados na nuvem. Diante disso, propõe-se neste trabalho, a elaboração de uma arquitetura multicamada para redução de dados baseada na computação em névoa (*Fog Computing*) denominada DaRkaM, acrônimo em inglês para (*Data Reduction Framework for Traffic Management*). O método consiste na utilização de uma estratégia de monitoramento capaz de realizar processos de redução de dados diretamente na borda da rede veicular. Na camada da nuvem, DaRkaM atua como um controlador central, analisando as posições geográficas dos veículos que são recebidas a partir de um fluxo de dados contínuo, os quais são utilizados nos processos de monitoramento e gestão do tráfego. Na borda da rede, um módulo de redução de dados implementado no framework DaRkaM foi projetado para hospedar diferentes estratégias de monitoramento de tráfego. Esta arquitetura permite a realização de análises comparativas envolvendo tanto a utilização de algoritmos baseados em amostragem simples, como o uso de algoritmos de agrupamento, nos quais os processos de redução de dados são estruturados com base na similaridade dos objetos analisados. Os resultados demonstraram que a utilização de algoritmos baseados em agrupamento, hospedados no núcleo de redução de dados do framework DaRkaM, alcançaram uma significativa redução no custo de comunicação, especialmente em cenários sobrecarregados, além de manterem alta precisão no monitoramento e detecção de congestionamento de tráfego.

**Palavras-chave:** Redução de Dados, Agrupamentos, Aprendizado de Máquina, Computação em Névoa, Redes Ad Hoc Veiculares, Sistemas de Transporte Inteligente.



## ABSTRACT

The Intelligent Transport Systems (ITS) has a function to analyze the flow of vehicles on highways in order to identify any traffic anomalies, ensuring greater efficiency during the decision-making process. These systems can be based on an Ad-Hoc vehicular network (VANETs) able to integrate the elements of urban space through a distributed communication system. Similarly, ITS applications require constant monitoring of roads, and such monitoring aims to analyze, among other aspects, the variation of vehicular density over time. In general, this process occurs by means of the periodic sending of situational data from the mobility environment to the cloud. Consequently, the data sets are sent at high frequency to the cloud, forming a continuous data flow that should be processed in a real-time context. However, this dynamic implies a progressive increase in the communication cost, in reason of the volume of data transferred in the link between the fog and the cloud, increasing the risks of overload beyond increasing the latency during requests for services made available in the cloud. Therefore, this work proposes the development of a multilayer architecture for data reduction based on *Fog Computing* called DaRkaM, acronym in English for (*Data Reduction Framework for Traffic Management*). The strategy consists of using a monitoring model able to perform data reduction processes directly at the edge of the vehicular network. At the cloud layer, DaRkaM acts as a central controller, analyzing the geographic positions of vehicles that are received from a continuous data flow. These data are used to monitor and perform the traffic management processes addressed in this proposal. At the edge network, a data reduction module was designed to host different traffic monitoring strategies. This architecture favors the comparative analysis among different approaches, ranging from the use of algorithms based on simple sampling to clustering algorithms, in which the data reduction processes are structured based on the number of clusters. The results showed that the use of cluster-based algorithms, hosted in the data reduction core of the DaRkaM framework, is able to achieve a great reduction in communication cost, especially in overloaded scenarios, beyond achieving high accuracy in monitoring and detecting traffic congestion.

**Keywords:** Data Reduction, Clusterization, Machine Learning, Fog Computing, Ad Hoc Vehicular Networks, Intelligent Transportation System.



# SUMÁRIO

<b>Capítulo 1—Introdução</b>	<b>1</b>
1.1 Hipótese . . . . .	5
1.2 Questão de Pesquisa . . . . .	5
1.3 Objetivos . . . . .	5
1.4 Objetivos Específicos . . . . .	5
1.5 Principais Contribuições . . . . .	6
1.6 Estrutura deste trabalho . . . . .	6
<b>Capítulo 2—Fundamentação Teórica</b>	<b>9</b>
2.1 Computação em Névoa . . . . .	9
2.1.1 Arquitetura da Névoa . . . . .	10
2.2 Estratégias de Redução de Dados . . . . .	13
2.3 Redes Ad Hoc Veiculares . . . . .	15
2.3.1 O Protocolo IEEE 802.11 para Redes VANET . . . . .	15
2.3.2 Frequência de Transmissão . . . . .	17
2.4 Simuladores de Tráfego e Conectividade . . . . .	17
2.5 Classificação do Tráfego por Nível de Serviço - LOS . . . . .	20
2.6 Fluxo de Dados Contínuos . . . . .	22
2.7 Estratégias de Agrupamentos Baseados em Densidade . . . . .	24
2.7.1 A Técnica de Agrupamento Espacial Baseada na Densidade Aplicada com Ruído (DBSCAN) . . . . .	25
2.7.2 Ordering Points to Identify the Clustering Structure (OPTICS) . . . . .	28
2.7.3 Hierarchical Density-based Spatial Clustering of Applications with Noise (HDBSCAN) . . . . .	32
2.7.4 X-Means . . . . .	35
2.8 K Shortest Path . . . . .	36
2.9 Considerações Finais . . . . .	37
<b>Capítulo 3—Trabalhos Relacionados</b>	<b>39</b>
3.1 Detecção de congestionamento em Sistemas de Transporte Inteligente . . . . .	39
3.2 Sistemas de transporte inteligente baseados em mídias sociais . . . . .	41
3.3 Predições no tráfego utilizado sistemas híbridos . . . . .	43
3.4 Técnicas de agrupamento aplicadas ao cenário urbano . . . . .	45
3.5 A redução de dados e os custos de comunicação . . . . .	47
3.6 Análise da Proposta em relação à Literatura Relacionada . . . . .	49

3.7	Considerações Finais . . . . .	52
<b>Capítulo 4—Data Reduction Framework for Traffic Management (DaRkaM)</b>		<b>53</b>
4.1	Aspectos Conceituais . . . . .	53
4.2	Arquitetura DaRkaM . . . . .	54
4.3	Fluxo de dados entre as camadas . . . . .	56
4.4	Simuladores Utilizados na Camada de Mobilidade . . . . .	57
4.5	Camada de Mobilidade - VANET . . . . .	58
4.6	Camada de Névoa . . . . .	59
4.6.1	Módulo de Monitoramento . . . . .	60
4.6.1.1	Algoritmo Baseline . . . . .	61
4.6.1.2	1-TO-2 . . . . .	62
4.6.1.3	Random . . . . .	63
4.6.1.4	Limite . . . . .	64
4.6.1.5	DBSCAN . . . . .	65
4.6.1.6	OPTICS . . . . .	68
4.6.1.7	X-Means . . . . .	69
4.6.1.8	HDBSCAN . . . . .	71
4.6.2	Level of Service - LOS . . . . .	72
4.6.3	Módulo de Roteamento . . . . .	73
4.7	Camada da Nuvem . . . . .	75
4.8	Considerações do Finais . . . . .	77
<b>Capítulo 5—Metodologia dos Experimentos e Análise dos Resultados</b>		<b>79</b>
5.1	Configurações do Cenário e Simulações . . . . .	79
5.2	Variações de Velocidade e Tendências . . . . .	82
5.3	Aplicando a classificação por nível de serviço (LOS) em todo o cenário . . . . .	89
5.4	Análise da Acurácia . . . . .	93
5.5	Coeficiente de Silhueta . . . . .	97
5.6	Analisando o Uso dos Recursos de Rede . . . . .	101
5.6.1	Uso da Rede e Custos de Comunicação . . . . .	101
5.6.2	Latência . . . . .	105
5.6.3	Pacotes Trafegados . . . . .	106
5.6.4	Throughput . . . . .	107
5.7	Volume de Dados Gerados . . . . .	108
5.8	Processo de Re-roteamento e tomada de decisão . . . . .	110
5.9	Discussão e Conclusão . . . . .	112
5.10	Considerações Finais . . . . .	115
<b>Capítulo 6—Conclusões</b>		<b>117</b>
6.1	Trabalhos Futuros . . . . .	118
6.2	Publicações . . . . .	119

## LISTA DE FIGURAS

2.1	Arquitetura da Computação em Névoa, adaptado de (MUKHERJEE; SHU; WANG, 2018) . . . . .	11
2.2	Probabilidade de entrega de beacon <i>vs</i> densidade (NOORI; OLYAEI, 2013). . . . .	17
2.3	Arquitetura Integrada dos Simuladores . . . . .	19
2.4	Visão dos Simuladores (SUMO, 2019) e (OMNET++, 2019) em execução . . . . .	19
2.5	Relação entre a taxa de velocidade x capacidade da rodovia (BOARD, 1997) . . . . .	21
2.6	Representação conceitual do funcionamento das Janelas Temporais . . . . .	24
2.7	Relação do raio de vizinhança. Adaptado de (CASSIANO, 2015) . . . . .	26
2.8	Relação do raio de vizinhança, adaptado de (CASSIANO, 2015) . . . . .	27
2.9	Métodos de Agrupamento DBSCAN (GIACOUMIDIS et al., 2019) . . . . .	28
2.10	Diferentes densidades, adaptado de (ANKERST et al., 1999) . . . . .	29
2.11	OPTICS Terminology (THILAGAMANI; JAYANTHILADEVI; ARUNKUMAR, 2018) . . . . .	29
2.12	Representação das definições para um $Min_{pts} = 4$ .(ANKERST et al., 1999) . . . . .	30
2.13	Medidas de conectividade. Adaptado de (ANKERST et al., 1999) . . . . .	31
2.14	$K = 8$ . As regiões de maior densidade (esquerda) têm um raio menor, enquanto as regiões de menor densidade têm um raio maior. Adaptado de (FILIPE; MOURA, 2021) . . . . .	33
2.15	Minimum Spanning Tree - MST de todos os pontos conectados por sua distância de alcance mútuo (MCINNES; J; S., 2016) . . . . .	34
2.16	Árvore HDBSCAN (MCINNES; J; S., 2016) . . . . .	35
3.1	Análise temporal dos avanços desta pesquisa . . . . .	51
4.1	Elementos Arquiteturais que Compõem o Framework DaRkaM . . . . .	54
4.2	Processamento do Fluxo de Dados Contínuo no <i>Framework</i> DaRkaM . . . . .	56
4.3	Modelo de Comunicação Veículo / Infraestrutura . . . . .	59
4.4	Mapeamento dos clusteres utilizando o algoritmo DBSCAN em fluxo de dados contínuo. . . . .	67
4.5	Classificação LOS como base para a distribuição de rotas alternativas . . . . .	76
5.1	Recorte da Malha Viária de Cologne, Alemanha. Cenário do Ambiente de Simulação . . . . .	81
5.2	Segmentos Selecionados para Monitoramento da Velocidade no Tráfego . . . . .	82
5.3	Análise da Velocidade ( $km/h$ ) no Segmento (1) . . . . .	83
5.4	Análise da Velocidade ( $km/h$ ) no Segmento (2) . . . . .	84
5.5	Análise da Velocidade ( $km/h$ ) no Segmento (3) . . . . .	85

5.6	Análise do Level of Service (LOS) no Segmento (1)	86
5.7	Análise do Level of Service (LOS) no Segmento (2)	87
5.8	Análise do Level of Service (LOS) no Segmento (3)	88
5.9	Detecção de congestionamento de tráfego com densidade de 100 veículos/ $km^2$	90
5.10	Detecção de congestionamento de tráfego com densidade de 200 veículos/ $km^2$	91
5.11	Detecção de congestionamento de tráfego com densidade de 300 veículos/ $km^2$	92
5.12	Aplicação do <i>Root-mean-square deviation</i> (RMSE) para as densidades de 100, 200 e 300 veículos	93
5.13	Análise da dispersão dos dados processados aplicando uma densidade de 100 veículos/ $km^2$	94
5.14	Análise da dispersão dos dados processados aplicando uma densidade de 200 veículos/ $km^2$	95
5.15	Análise da dispersão dos dados processados aplicando uma densidade de 300 veículos/ $km^2$	96
5.16	Análise do Coeficiente Silhueta para 100, 200 e 300 veículos/ $km^2$	97
5.17	Coeficiente silhueta x <i>clusters</i> formados em um cenário com 100 veículos	98
5.18	Coeficiente silhueta x <i>clusters</i> formados em um cenário com 200 veículos	99
5.19	Coeficiente silhueta x <i>clusters</i> formados em um cenário com 300 veículos	100
5.20	Comparativo da Largura de Banda e a Quantidade de Veículos.	101
5.21	Estimativa de Custos de Armazenamento	104
5.22	Latência 100, 200 e 300 veículos/ $km^2$ .	105
5.23	Taxa de envio de pacotes 100, 200 e 300 veículos/ $km^2$ .	106
5.24	Monitoramento do Throughput para 100, 200 e 300 veículos/ $km^2$ .	107
5.25	Média da Redução Durante os Experimentos	108
5.26	Relação entre o tamanho do arquivo e a densidade veicular	109
5.27	Velocidade média das rodovias durante a simulação	111
5.28	Tempo de viagem	112

## LISTA DE TABELAS

2.1	Pacote Beacon . . . . .	16
3.1	Matriz de Trabalhos Relacionados . . . . .	50
4.1	Classificação LOS/HCM . . . . .	72
5.1	Parâmetros de Simulação . . . . .	80
5.2	Classificação ( $w_{ij}$ ) das Condições de Tráfego . . . . .	89
5.3	Tabela Comparativa da Largura de Banda e a Quantidade de Veículos. .	103
5.4	Análise comparativa dos resultados por algoritmos e métricas aplicadas .	113



## LISTA DE SIGLAS

<b>AHP</b>	<i>Analytical Hierarchy Process</i> . . . . .	45
<b>ANP</b>	Agência Nacional do Petróleo, Gás Natural e Biocombustíveis . . . . .	1
<b>BIC</b>	<i>Bayesian Information Criterion</i> . . . . .	35
<b>BS</b>	<i>Base Station</i> . . . . .	59
<b>CCH</b>	<i>Control Channel</i> . . . . .	16
<b>DaRkaM</b>	<i>Data Reduction Framework for Traffic Management</i>	
<b>DBSCAN</b>	<i>Density-Based Spatial Clustering of Applications with Noise</i> . . . . .	25
<b>DSRC</b>	<i>Dedicated Short-Range Communication</i> . . . . .	16
<b>DTMS</b>	<i>Dynamic Traffic Management System</i> . . . . .	41
<b>FCD</b>	Fluxo Contínuo de Dados . . . . .	22
<b>FDC</b>	<i>Floating Car Data</i> . . . . .	43
<b>GPS</b>	<i>Global Positioning System</i> . . . . .	40
<b>HCM</b>	<i>Highway Capacity Manual</i> . . . . .	20
<b>HDBSCAN</b>	<i>Hierarchical Density-based Spatial Clustering of Applications with Noise</i> . . . . .	25
<b>IDC</b>	<i>International Data Corporation</i> . . . . .	2
<b>IoT</b>	<i>Internet of Things</i> . . . . .	10
<b>IPEA</b>	Instituto de Pesquisa Econômica Aplicada . . . . .	1
<b>ITS</b>	<i>Intelligent Transport Systems</i> . . . . .	3
<b>KSP</b>	<i>K-shortest paths</i> . . . . .	9
<b>LOS</b>	<i>Level of Service</i> . . . . .	6
<b>LSTM</b>	<i>Long Short-Term Memory</i> . . . . .	44
<b>MAV</b>	<i>Maximum Value in Each Grid</i> . . . . .	44
<b>MFO</b>	<i>Moth-Flame Optimization</i> . . . . .	46
<b>MRS</b>	<i>Message Recovery Services</i> . . . . .	45
<b>OBD</b>	<i>On Board Unit On-Board Diagnostic</i> . . . . .	40
<b>OBU</b>	<i>On Board Unit</i> . . . . .	40
<b>OFDM</b>	<i>Orthogonal Frequency-Division Multiplexing</i> . . . . .	16

<b>OPEX</b>	<i>Operational Expenditure</i>	4
<b>OPTICS</b>	<i>Ordering points to identify the clustering structure</i>	25
<b>QOS</b>	<i>Quality of Service</i>	105
<b>RMSE</b>	<i>Root Mean Square Error</i>	44
<b>RSU</b>	<i>Road Side Unit</i>	4
<b>SCH</b>	<i>Service Channels</i>	16
<b>SUMO</b>	<i>Simulation of Urban Mobility</i>	68
<b>V2I</b>	<i>Vehicle-to-Infrastructure</i>	3
<b>V2V</b>	<i>Vehicle-to-Vehicle</i>	15
<b>VANET</b>	<i>Vehicular Ad Hoc Network</i>	3
<b>VEINS</b>	<i>Vehicle in Network Simulation</i>	79
<b>WAVE</b>	<i>Wireless Access in Vehicular Environments</i>	3

## INTRODUÇÃO

A modernização dos grandes centros urbanos ao redor do globo implica na superação de novos e significativos desafios de gestão como o controle da poluição do ar, aumento do consumo de energia, congestionamento de trânsito, além de problemas socioeconômicos e estruturais, os quais impactam diretamente no sistema de bem-estar social. No enfrentamento dessas questões, aplicações com ênfase na gestão da mobilidade urbana vêm obtendo significativa relevância, estas aplicações podem contribuir para melhorias no fluxo de tráfego veicular, oferecendo apoio em situações emergenciais, além de viabilizar a construção de novos serviços relacionados à mobilidade (MENG et al., 2022) (CHAKRABORTY, 2022), (HOU et al., 2016), (ZHENG; GAO; LIN, 2013), (BATTY et al., 2012).

Zheng et al. (2014), definiu a computação urbana como o processo de aquisição, integração e análise de grandes volumes de dados heterogêneos, dispostos em um espaço urbano e gerados por diversas fontes, tais como: sensores, dispositivos, veículos, prédios e humanos, cujo objetivo é lidar com as principais questões enfrentadas nas cidades. Para tanto, a computação urbana abarca em sua estrutura conceitual um conjunto de aplicações que vão desde o planejamento do espaço urbano, cuidados com o meio ambiente até o gerenciamento em sistemas de transporte rodoviários, sendo este último um dos principais desafios de mobilidade a ser enfrentado por essas tecnologias.

Os sistemas de transporte rodoviário cumprem um papel essencial para mitigar alguns dos desafios relacionados à mobilidade urbana, uma das áreas mais impactadas pelo progressivo processo de urbanização. No Brasil, de acordo com o Instituto de Pesquisa Econômica Aplicada (IPEA), moradores dos grandes centros urbanos gastam em média duas horas para se locomoverem a cada dia. Um estudo realizado pelo Instituto de Pesquisa Ipsos identificou que 70% dos entrevistados relataram sofrer com severos congestionamentos todos os dias (GALINDO; CORREIA; NETO, 2019). Adicionalmente, um estudo realizado pela Agência Nacional do Petróleo, Gás Natural e Biocombustíveis (ANP), identificou um aumento nos volumes de combustíveis vendidos pelas distribuidoras nos grandes centros, alcançando marcas que superam os anos de 2019 e 2020. Em 2021, as vendas de gasolina no país somaram cerca de 39,3 bilhões de litros, ou 9,7% a mais do

que em 2020, confirmando, em certa medida, a volta da mobilidade urbana, passado o auge da pandemia (INFOMONEY, 2022).

Embora o fluxo de tráfego tenha aumentado significativamente nos últimos anos, estes números foram reduzidos drasticamente nos anos de 2020 e 2021 (SCHRANK et al., 2021). Isso ocorreu em função da pandemia do COVID19 que assolou o mundo ao longo destes dois anos. Todavia, com a redução no número de casos observada nos meses mais recentes, analistas de transporte vêm observando um aumento progressivo e acelerado dos níveis de congestionamento de tráfego, o que sugere a necessidade de identificar mecanismos de gerenciamento de tráfego mais eficientes, capazes de suportar uma distribuição mais organizada e eficiente do fluxo de tráfego, uma análise essencial para lidar com o congestionamento de tráfego pós-pandemia, principalmente nos grandes centros urbanos (TIMES, 2021; ITDP, 2021; CIVICWELL, 2021).

Diante disso, a computação urbana procura lidar com o desafio da gestão do tráfego por meio do uso de tecnologias de detecção e infraestruturas de computação em larga escala. Nesse contexto, tal infraestrutura tem a função compor um ambiente distribuído capaz de apoiar etapas como a seleção de dados, processamento, análise e tomada de decisão a partir dos grandes volumes de dados gerados pelos elementos contidos no espaço urbano. Os desafios desta abordagem vão desde o uso dos recursos para estabelecer a correta comunicação entre estes elementos, até a realização de processamentos e ações que contribuam para garantir a eficiência destes espaços.

Ações de monitoramento, em um sistema de transporte rodoviário, requerem uma infraestrutura capaz de lidar com estes grandes conjuntos de dados, no entanto, o crescimento do número de dispositivos em função da modernização dos centros urbanos vem acentuando, de forma gradual e contínua, este processo. Uma pesquisa realizada pelo *International Data Corporation* (IDC) atenta para o fato de que o número de dispositivos gerais, conectados, pode alcançar 80 bilhões até 2025, com um crescimento de mercado na ordem de 28% apenas entre 2018 e 2020 (IDC, 2019). Desse total, estima-se 110 milhões de carros conectados com cerca de 5,5 bilhões de sensores, além de 1,2 milhão de residências conectadas com 200 milhões de sensores (IDC, 2016). Analogamente, Gino (2006) prevê que em 2023, quase 70% dos novos veículos em todo o mundo (90% nos Estados Unidos) terão uma interface de conectividade integrada a uma rede móvel, essa conectividade permitirá o tráfego de dados veiculares provenientes de gateways, softwares ou sensores (IDC, 2019). Acredita-se que esse grau de conectividade entre os mais variados tipos de dispositivos levará a um crescimento exponencial de dados, maximizando os custos de comunicação na rede e a complexidade computacional para orquestração de dados (PEIXOTO; GENEZ; BITTENCOURT, 2022).

Nesse contexto, cabe destacar que esse aumento progressivo no volume de dados traz impactos não só relativos à conectividade, mas também implica no enfrentamento de novos desafios relacionados à orquestração e gerenciamento destes dados, aumentando a complexidade em áreas como armazenamento, processamento, análise e extração de conhecimento, impondo desafios no que diz respeito à redução da latência e elevação dos custos de comunicação na rede, especialmente em função do volume de dados trocados no enlace entre a névoa e a nuvem (DESAI, 2018; MUKHERJEE; SHU; WANG, 2018; PEIXOTO; GENEZ; BITTENCOURT, 2022).

Um meio para lidar com esses grandes conjuntos de dados produzidos no cenário urbano são os Sistemas de Transporte Inteligente ou *Intelligent Transport Systems (ITS)*. Parte da computação urbana, essas aplicações têm a função de aprimorar o desempenho do tráfego em rodovias monitoradas. Aplicações ITS têm motivado a oferta de uma nova geração de serviços virtuais, entre eles destacam-se aplicativos de viagens que incorporam serviços relacionados à oferta de caronas, aplicativos governamentais para transporte público, aplicativos de táxi, além de serviços de viagens compartilhadas (CERQUEIRA et al., 2018).

Importante destacar que todos esses aplicativos têm em comum a necessidade de um subsistema capaz de oferecer informações relativas ao fluxo de tráfego nas rodovias com certo grau de acurácia e de forma continuada. Essas informações, em constante atualização, podem auxiliar no planejamento das rotas dos veículos e, como consequência, garantir maior eficiência em aspectos como redução do tempo de viagem, estratégias para evitar colisões, redução no uso de combustível, além de atuar na redução de emissões atmosféricas (RODRIGUEZ; FERREIRA; POCO, 2022; SHAH et al., 2022). De maneira geral, as informações obtidas por esse subsistema são processadas e podem ainda retroalimentar esse mesmo subsistema, construindo novas análises que podem gradativamente aumentar a sua eficiência na oferta dos serviços.

Para realizar esta tarefa, os sistemas de transporte inteligente requerem uma estrutura de comunicação capaz de conectar todos os elementos dispostos no espaço urbano. Nos cenários de mobilidade, esta conectividade pode ser proporcionada pelas *Vehicular Ad Hoc Network (VANET)* (SOUZA et al., 2018) que utilizam a comunicação sem fio em ambientes veiculares *Wireless Access in Vehicular Environments (WAVE)* operando sob um certo protocolo, a exemplo do IEEE 802.11p (SCHMIDT et al., 2010). A comunicação constante entre os elementos do cenário urbano torna-se essencial para obter um entendimento acerca da dinâmica do tráfego veicular, sendo esse o primeiro passo para a implementação de estratégias que maximizem a qualidade no fluxo nas rodovias monitoradas.

Na literatura, observam-se com certa recorrência, abordagens que enfatizam o agrupamento de grandes conjuntos de dados trafegados nessas redes veiculares para formar uma visão global do estado do tráfego, enviando esses conjuntos para processamento em um servidor centralizado baseada em uma arquitetura de computação em nuvem. Autores que corroboram com essa abordagem (GUPTA et al., 2022; WAHID; RAO; GOEL, 2019; BILAL et al., 2018; MUKHERJEE; SHU; WANG, 2018; BASAGNI et al., 2013; TANIZAKI; WOLFSON, 2007; Kerner et al., 2005) argumentam que a comunicação entre veículos e a infraestrutura ou *Vehicle-to-Infrastructure (V2I)* tende a considerar o sistema de monitoramento de tráfego como um modelo arquitetural. Nesse modelo, veículos do cenário de mobilidade enviam periodicamente grandes quantidades de dados para a nuvem, e essa, por sua vez, atua como um controlador central que reconhece a posição geográfica de todos os veículos, adquirindo dados em fluxo com o fim de realizar a detecção de congestionamento de tráfego e eventuais intervenções relacionadas à mobilidade.

Por outro lado, áreas metropolitanas podem alcançar até milhões de veículos conectados à nuvem e esse volume de dados de tráfego traz como consequência o aumento

progressivo dos custos de comunicação, um cenário que pode representar uma barreira para expansão dos sistemas ITS. Analogamente, embora as redes 5G sejam provisionadas para suportar o crescimento exponencial do tráfego, o uso regular da rede LTE / 5G em um ambiente urbano, combinado com o crescimento dos dados da VANET, também podem levar à sobrecargas dos canais de comunicação que integram o ambiente de mobilidade e a nuvem. Assim, o paradigma da computação em névoa (*Fog Computing*) emerge como um meio de mitigar tais restrições. A abordagem consiste em estender os recursos computacionais disponíveis na nuvem para a borda da rede com o objetivo de apoiar a troca de informações, reduzindo a latência em função das distâncias entre os dispositivos e uma certa unidade de processamento (BONOMI et al., 2012).

Diante disso, propõe-se neste trabalho a elaboração de um *framework* de redução de dados que utiliza técnicas de agrupamento para extração de subconjuntos representativos, capazes de apoiar na detecção de congestionamentos e gerenciamento do tráfego utilizando uma fração dos dados trafegados pela infraestrutura. O *framework* DaRkaM, acrônimo em inglês para *Data Reduction Framework for Traffic Management* tem por objetivo reduzir os custos de comunicação inerentes às atividades de gerenciamento do tráfego, utilizando uma abordagem baseada em névoa como extensão da nuvem para formar uma arquitetura multicamada capaz de atuar em um cenário de fluxo de dados contínuo.

A arquitetura proposta ao longo deste trabalho baseia-se em três camadas distintas, porém complementares, a saber *a)* VANET ou Camada de mobilidade, representando os requisitos de comunicação e conectividade encontrados no espaço urbano; *b)* Camada da névoa, na qual ocorrem os processos de redução de dados e por fim *c)* Camada da nuvem, na qual ocorrem os processos de tomada de decisão baseados no monitoramento das rodovias. Cada uma das camadas, atua em um contexto de fluxo de dados contínuo no qual, dados produzidos pela camada de mobilidade são processados pelas camadas adjacentes do modelo. Este processamento é realizado por meio das unidades de processamento distribuídas nas rodovias, denominadas por *Road Side Unit (RSU)*. Dados oriundos dessas unidades de processamento são submetidos à camada da névoa na qual são realizados os processos de redução de dados utilizando estratégias baseadas em agrupamento.

Os dados resultantes das operações de agrupamento configuram a parcela mais representativa dos conjuntos processados, criando um subconjunto capaz de apoiar na análise e identificação do estado atual das rodovias em termos de tráfego e densidade veicular. Estes subconjuntos de dados também são utilizados nos processos de tomada de decisão, apoiando a realização de intervenções no cenário urbano capazes de reduzir o tempo de viagem através da utilização de estratégia de roteamento implementadas sob essa fração do conjunto de dados original. A redução de dados contribui tanto para ampliar a capacidade de processamento dos dados, quanto para reduzir os custos gerais de operação.

Os efeitos práticos relativos à estratégia de redução dos dados proposta nesta tese podem ser analisados não só sob a perspectiva da redução da complexidade do gerenciamento, como também a partir da redução dos custos relativos à *Operational Expenditure (OPEX)*. Nesse contexto, duas relevantes classes de custo podem ser positivamente impactadas, são elas: os custos relativos ao armazenamento de dados e os custos de comunicação, os quais figuram entre os principais benefícios da presente proposta. As próximas seções abordam detalhes acerca do desenvolvimento desta proposta delineando

a hipótese, questão de pesquisa e objetivos estabelecidos na elaboração desse trabalho.

## 1.1 HIPÓTESE

Um sistema de transporte inteligente, baseado na computação em névoa, pode se utilizar de técnicas de agrupamento para apoiar a redução de dados, diminuindo os custos de comunicação sem reduzir a acurácia das análises para gerenciamento do fluxo de tráfego.

## 1.2 QUESTÃO DE PESQUISA

Baseado na hipótese descrita na seção anterior, foi definida a seguinte questão de pesquisa que norteou o desenvolvimento deste trabalho: *Um framework de redução de dados, baseado em computação em névoa, (Fog Computing) utilizando algoritmos de agrupamento em um contexto de fluxo de dados contínuo, pode contribuir de forma efetiva para maximizar a eficiência no fluxo de tráfego e, simultaneamente, reduzir os custos de comunicação?*

## 1.3 OBJETIVOS

O objetivo desse trabalho é propor e implementar o *framework* de redução de dados DaR-kaM, capaz de atuar como um sistema de transporte inteligente, utilizando o paradigma da computação em névoa para mitigar os desafios relacionados à latência reduzida e o aumento progressivo nos custos de comunicação durante as ações de identificação de congestionamento, monitoramento e gestão do tráfego. A seguir, alguns objetivos específicos são apresentados.

## 1.4 OBJETIVOS ESPECÍFICOS

- Sincronizar um conjunto de simuladores para projetar um ambiente veicular capaz de reproduzir desafios do espaço urbano no qual seja possível realizar os experimentos e validações da presente proposta;
- Estruturar um padrão de troca de mensagens com o fim de padronizar o recebimento de dados em fluxo contínuo e o tráfego desses dados entre as camadas do modelo;
- Implementar as estratégias de redução de dados na névoa utilizando o conceito de janelas temporais;
- Implementar os modelos de classificação com base na coleta e análise dos dados recebidos na névoa;
- Definir um padrão de troca de informações entre a nuvem e a camada de mobilidade com o fim de enviar sugestões de roteamento para os veículos com base no estado atual das rodovias;
- Realizar um planejamento de experimento utilizando replicações para mensurar a influência dos algoritmos na acurácia e tomada de decisão em ambiente de simulação de tráfego;

## 1.5 PRINCIPAIS CONTRIBUIÇÕES

As principais contribuições desta tese estão relacionadas à redução de dados em redes veiculares, permitindo melhor gerenciamento, redução da complexidade e dos custos de comunicação destes ambientes. A seguir, algumas características que se destacaram durante o desenvolvimento desta tese:

1. **Arquitetura Proposta - *Framework* DaRkaM:** Trata-se de uma arquitetura desenvolvida para garantir a seleção, processamento e distribuição do fluxo de dados contínuo recebido a partir da rede veicular;
2. **Estratégias de redução de dados:** Adaptação dos algoritmos **DBSCAN**, **X-Means**, **OPTICS** e **HDBSCAN** para operar em um contexto de fluxo de dados contínuo, permitindo a identificação de pontos de retenção a partir de um percentual de dados contidos nos agrupamentos identificados pelas diferentes abordagens;
3. **Redução dos custos de comunicação:** As estratégias de redução de dados culminam na consequente redução dos custos relacionados ao OPEX, impactando positivamente em dois fatores principais: redução no armazenamento de dados e redução dos custos de comunicação;
4. **Classificação em fluxo de dados contínuo:** Estabelecimento de um mecanismo de classificação contínua do estado do tráfego, baseado no monitoramento dos dados obtidos a partir da VANET utilizando a escala *Level of Service* (LOS);
5. **Mecanismo de re-roteamento integrado:** Fundamentado nos dados de classificação obtidos em tempo real, foram implementados os mecanismos de análise e re-roteamento dos veículos com base no estado atual do tráfego.
6. **Submissões:** Realização de submissões parciais e completa desta pesquisa com o fim de validar a presente proposta junto à comunidade acadêmica.

## 1.6 ESTRUTURA DESTE TRABALHO

Além da introdução, este trabalho está dividido da seguinte forma: no **Capítulo 2** – Fundamentação Teórica, serão abordados os conceitos fundamentais utilizados nessa pesquisa. Nesse capítulo, o leitor será apresentado a conceitos como sistemas de transporte inteligente, passando por algoritmos de agrupamento, além de fundamentos relacionados à redes veiculares, necessários ao entendimento da proposta. **Capítulo 3** – Análise dos Trabalhos Correlatos: Com base nos resultados obtidos através da revisão sistemática da literatura, esse capítulo procura apresentar uma análise dos trabalhos selecionados. Nesse contexto, o capítulo enseja as discussões em torno das redes veiculares e alternativas para redução de dados na névoa. **Capítulo 4** – Proposta de Pesquisa: Aqui, são apresentados os detalhes do framework proposto nesse trabalho, bem como os ambientes e cenários utilizados durante os experimentos. **Capítulo 5** – Metodologia e Resultados, no qual são apresentados os resultados alcançados, a partir da condução de experimentos,

coleta e análise dos dados obtidos. **Capítulo 6** – Considerações: Esse capítulo discute as considerações finais desse trabalho. Aqui, são relatadas as contribuições alcançadas nessa pesquisa, publicações de artigos submetidos, além de sugestões para pesquisas futuras.



## FUNDAMENTAÇÃO TEÓRICA

Este capítulo aborda os principais conceitos utilizados na presente proposta com o fim de auxiliar no entendimento sobre as tecnologias, processos e métodos empregados na elaboração deste trabalho. O conteúdo deste capítulo está organizado da seguinte forma: Na Seção 2.1, são analisados os meandros da computação em névoa e como este paradigma pode contribuir para mitigar problemas relacionados à latência, sobrecarga de dados e custo de comunicação. Na Seção 2.2, são apresentados os conceitos fundamentais da redução de dados e seus reflexos nos custos de comunicação. Posteriormente, na Seção 2.3, são analisadas as redes veiculares ad hoc ou VANETs. Na Seção 2.4 são analisados as ferramentas de simulação utilizadas nesta pesquisa. Na Seção 2.5 são analisados os conceitos da classificação por nível de serviço e como ela pode ser aplicada nas medições do estado do tráfego. Na Seção 2.6, são analisados os desafios relacionados ao desenvolvimento de aplicações para fluxo de dados contínuos e os aspectos gerais dessa abordagem. Na Seção 2.7, são apresentados os aspectos principais relacionados ao aprendizado de máquina supervisionado em VANETS. Por fim, na Seção 2.8, são abordados conceitos de re-roteamento, analisando as abordagens baseadas em *K-shortest paths* (KSP) para definições de rotas e intervenções no cenário urbano.

### 2.1 COMPUTAÇÃO EM NÉVOA

O surgimento da Internet das Coisas desafia os modelos de computação tradicionais e centralizados como a computação em nuvem. Se por um lado o modelo “*pay-as-you-go*” oferece uma maneira eficiente de gerenciamento de dados e aplicações tradicionais, por outro, o crescimento contínuo do número de dispositivos conectados, associados à demanda por um modelo distribuído capaz de analisar fluxos permanente de dados, com baixa latência e baixo impacto nos custos de comunicação permanece um desafio, até mesmo para a nuvem (MUKHERJEE; SHU; WANG, 2018).

O paradigma da computação em névoa emerge como uma forma de mitigar tais desafios. A abordagem consiste, fundamentalmente, na extensão dos recursos computacionais de comunicação e armazenamento para a borda da rede, próximos aos dispositivos, atuando como uma alternativa para a construção de aplicações geograficamente distribuídas,

sensíveis à latência e ecossistemas *Internet of Things* (IoT) mais eficientes. Bonomi et al. (2012) definiu a computação em névoa como uma plataforma altamente virtualizada que fornece serviços de computação, armazenamento e comunicação entre os dispositivos finais e os centros de dados tradicionais de computação em névoa, localizados normalmente, mas não exclusivamente, na borda da rede. Entre as suas principais características estão a baixa latência, capacidade de atuar de forma geograficamente distribuída, heterogeneidade, interoperabilidade, escalabilidade e interações em tempo real (IORGA et al., 2018).

Diante disso, para cumprir os requisitos essenciais de um ambiente de computação em névoa, esta arquitetura deve contemplar um certo conjunto de características que tornam a névoa uma extensão não trivial da nuvem, a saber: *a)* Reconhecimento da localidade; *b)* Redução da latência; *c)* Distribuição geográfica; *d)* Suporte a redes de sensores em larga escala *e)* Suporte a grande número de nós; *f)* Suporte à computação móvel; *g)* Interações em tempo real; *h)* Heterogeneidade; *i)* Interoperabilidade; *j)* Análise de dados em tempo real. Tais características tornam a computação em névoa uma plataforma unificadora, na medida em que oferece a esta nova geração de serviços a capacidade de expansão no desenvolvimento de novas aplicações. Adicionalmente, a névoa também pode ser usada para reduzir a sobrecarga de dados enviados para a nuvem (ELBAMBY et al., 2014; BASTUG; BENNIS; DEBBAH, 2014; ELBAMBY; BENNIS; SAAD, 2017). Cabe destacar, todavia, que a computação em névoa não substitui a nuvem, trata-se de soluções diferentes e complementares, as quais são muitas vezes necessárias em conjunto.

A arquitetura distribuída, proposta pelo paradigma da computação em névoa, oferece uma sólida base para a construção de aplicações e serviços para cenários urbanos, permitindo o desenvolvimento de uma comunicação distribuída, com baixa latência e capaz de atuar de maneira eficiente na melhoria da qualidade de vida do cidadão. Analogamente, o desenvolvimento de aplicações distribuídas requer uma arquitetura igualmente distribuída, neste contexto, o planejamento dessa arquitetura deve considerar não só o planejamento do processamento de borda, mas também sua interação com os diferentes componentes distribuídos e a nuvem. A seção seguinte aborda o modelo arquitetural padrão contendo as principais características dessa abordagem.

### 2.1.1 Arquitetura da Névoa

Para entender como a computação em névoa é capaz de apoiar as soluções distribuídas e ecossistemas IoT, faz-se necessário entender a sua arquitetura e como esse paradigma estende os recursos da nuvem para a borda da rede, abrindo espaço para o desenvolvimento de um novo conjunto de aplicações e serviços distribuídos. Conforme já mencionado, o paradigma da computação em névoa atua como um elemento integrador, capaz de maximizar os recursos de comunicação por meio do processamento intermediário, gerenciando a troca de mensagens entre dispositivos e a nuvem. Para isso, uma arquitetura convencional da computação em névoa pode ser estruturada sobre um conjunto de camadas independentes que se integram para viabilizar a comunicação, processamento e integração entre os elementos presentes nessa rede. Uma representação conceitual do modelo arquitetural na névoa é ilustrada na Figura 2.1.

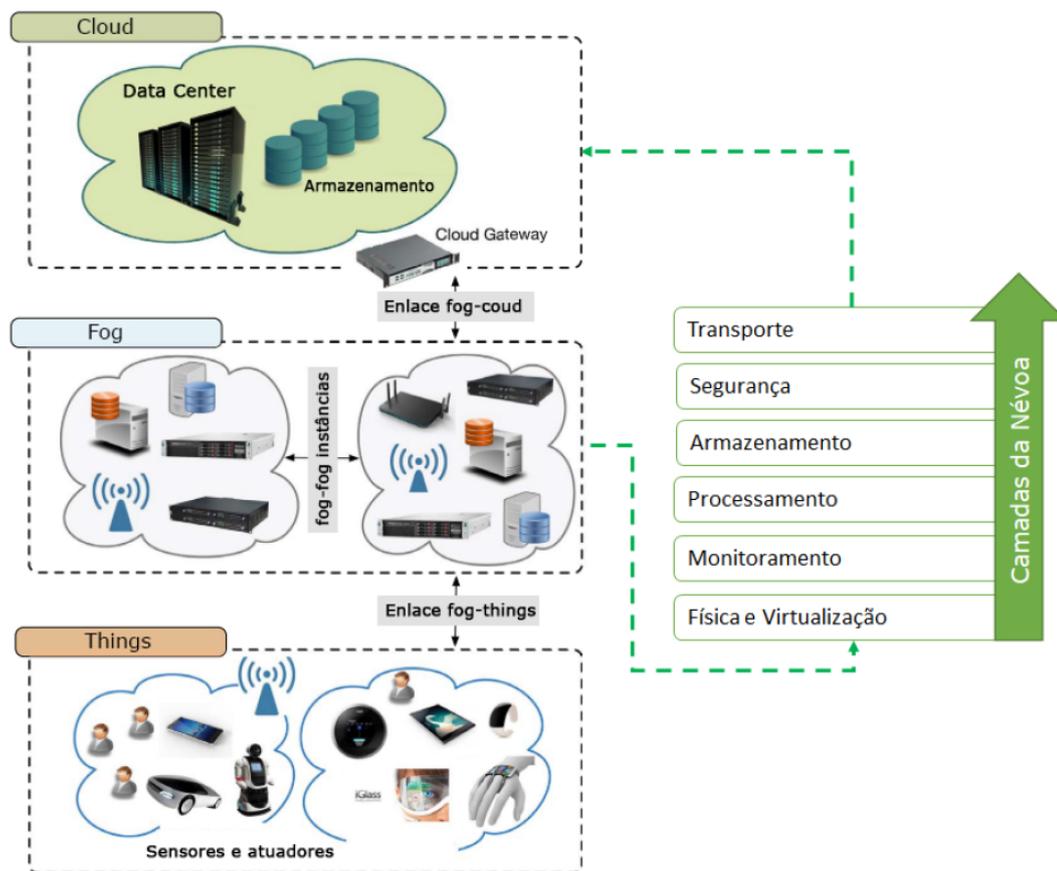


Figura 2.1: Arquitetura da Computação em Névoa, adaptado de (MUKHERJEE; SHU; WANG, 2018)

A Figura 2.1 apresenta um modelo da computação em névoa baseado em três macro camadas (SARKAR; CHATTERJEE; MISRA, 2018). Esta é a mais amplamente aceita representação arquitetural da névoa. Dentro de cada camada, outros modelos são estruturados de forma a operacionalizar as funcionalidades da arquitetura. Assim, esta estrutura está distribuída entre as seguintes camadas:

- **Camada de coisas:** nesta camada, estão sensores, atuadores e demais dispositivos conectados que enviam, periodicamente, informações para as camadas superiores.
- **Camada de névoa:** responsável por receber os dados da camada inferior. Esta camada possui elementos de conectividade que permitem o compartilhamento de recursos, armazenamento e processamento de forma colaborativa e distribuída.
- **Camada de nuvem:** camada composta, essencialmente, por servidores de nuvem em um modelo de *data center* tradicional. Esta camada possui alta capacidade de processamento e armazenamento.

Analisando mais detalhadamente a camada de computação em névoa, os autores (AAZAM; HUH, 2016) argumentam que esta pode ser estruturada em um conjunto de seis subcamadas que descrevem os processos que vão desde a recepção dos dados até o seu envio para a nuvem.

Em sua abordagem, os autores definiram as seguintes subcamadas:

1. **Camada física:** esta camada se utiliza de nós de borda para manter a conectividade com a camada de coisas;
2. **Camada de monitoramento:** responsável pelo monitoramento contínuo dos recursos, atendendo requisições e questões relacionadas ao consumo de energia;
3. **Camada de processamento:** responsável por realizar o processamento das entradas ou conjuntos de dados obtidos da camada inferior;
4. **Armazenamento temporário:** responsável por armazenar os dados, temporariamente, durante ou após as operações e processamentos;
5. **Camada de segurança:** responsável por garantir os aspectos de segurança na névoa;
6. **Camada de transporte:** se utiliza das interfaces *fog-cloud* para enviar os dados resultantes do processamento para a camada superior.

A arquitetura proposta por (AAZAM; HUH, 2016) e sua estruturação de camadas têm por fim dividir as responsabilidades que envolvem a comunicação e operacionalização de um ambiente de névoa. Esta estrutura preconiza um fluxo de atividades no qual cada camada opera de forma independente, de modo que o processamento tem início na camada física, onde sensores emitem sinais de comunicação com a infraestrutura; A camada de monitoramento aguarda tais requisições para encaminhá-las à camada subsequente; Os dados recebidos por estas requisições são processados na camada de processamento; A camada de armazenamento oferece suporte limitado para guarda e recuperação de dados por um certo período; A integridade da troca de mensagens é responsabilidade da camada de segurança e por fim, os dados são enviados para a nuvem através da camada de transporte.

A abordagem proposta por (AAZAM; HUH, 2016) norteou alguns aspectos deste trabalho, não só pela disposição arquitetural das camadas, abordagem utilizada no próprio *Framework* DaRkaM, mas principalmente, pelo mecanismo de comunicação entre os elementos presentes na rede. Embora o contexto da presente proposta esteja centrado na computação urbana e ambientes de mobilidade, uma arquitetura multicamada envolvendo a computação em névoa é necessária para intermediar as reações entre camadas inferiores (veículos e infraestrutura) com a camada de nuvem.

## 2.2 ESTRATÉGIAS DE REDUÇÃO DE DADOS

Tecnologias como a computação em névoa têm cumprido um importante papel ao permitir que recursos da computação em nuvem sejam estendidos até um local próximo aos dispositivos, com o fim de garantir a maior qualidade de serviço e confiabilidade em termos de latência e eventuais intermitências. Assim, o paradigma da computação em névoa tem trazido benefícios inegáveis para as arquiteturas de computação distribuída, apoiando a consolidação de conceitos como a internet das coisas e cidades inteligentes.

Entretanto, se por um lado os requisitos relacionados à baixa latência podem ser melhor assegurados apenas com a aplicação da computação em névoa, por outro, os custos computacionais da nuvem permanecem diretamente proporcionais ao crescimento da quantidade de dados trafegados na névoa. No contexto de *big data*, esta relação torna-se uma barreira para a popularização de certas soluções de sensoriamento como os sistemas de transporte inteligente e sistemas de tempo real.

Para exemplificar, as redes veiculares *ad hoc* consistem em importantes soluções para o contexto dos sistemas de transporte inteligente e representam, em última análise, o meio pelo qual os elementos dispostos em espaços urbanos são capazes de se comunicar entre si e com serviços de infraestrutura. No entanto, estimativas sugerem que serão mais de 150 milhões de carros conectados nos próximos anos e assumindo que cada carro pode gerar em média 30 TB de dados por dia (NING; HUANG; WANG, 2019), tem-se um cenário cada vez mais desafiador do ponto de vista da largura de banda operando sobre protocolos como o IEEE 802.11p.

Diante disso, as técnicas de redução de dados representam um meio para diminuir os impactos na correlação entre o aumento do fluxo de dados e a sua direta correspondência no aumento dos custos de comunicação. A rigor, existem duas técnicas principais para a implementação de estratégias de redução de dados na névoa, são elas: em *batch*, onde os dados são armazenados em uma área temporária para que sejam analisados integralmente depois de um certo tempo, ou a análise do *fluxo contínuo de dados*, no qual, conjuntos de dados são continuamente recebidos de uma infraestrutura urbana. O framework **DaRkaM**, proposto neste trabalho, utiliza uma abordagem baseada em fluxo de dados contínuo na camada da névoa.

Entre as técnicas de redução em fluxo de dados, pode-se destacar: *a) Amostragem de reservatório*, esta abordagem baseia-se na definição de uma amostragem aleatória obtida a partir de um dado conjunto, no qual o número de elementos é desconhecido (VITTER, 1985); *b) Descarte de cargas* (TATBUL et al., 2003), esta abordagem sugere a aplicação de métodos de processamento para identificar quando e onde reduzir e o quanto da carga pode ser reduzida e, por fim, têm-se a *c) Agregação de mensagens* (TATBUL et al., 2003) a qual se utiliza de técnicas baseadas em algoritmos de agrupamento para coletar mensagens enviadas pelos veículos. Essas técnicas enfatizam a formação de grupos *clusters* com o objetivo de selecionar apenas os elementos de maior representatividade em um certo conjunto de dados. Neste contexto, a estratégia pressupõe que veículos próximos em um congestionamento tendem a emitir mensagens similares e, portanto, estas poderiam ser detectadas pelas estratégias de agrupamento.

Algumas técnicas baseadas na seleção por amostragem simples, em um contexto de

fluxo de dados contínuo, encontradas na literatura são **1to2**, **Limite**, **Random**. Cada uma dessas técnicas se utiliza de um método para realizar o descarte de certos elementos, a fim de reduzir os dados antes de enviá-los para a nuvem. Algumas dessas estratégias foram implementadas nesta pesquisa com o fim de fornecer uma base comparativa entre as diferentes abordagens.

O algoritmo **1to2** utiliza uma estratégia baseada no descarte de 1 elemento para cada dois outros recebidos no fluxo de dados. Esta técnica promove uma redução de dados significativa frente ao conjunto de dados original, entretanto, estudos identificaram uma relevante perda de acurácia em certos cenários.

Analogamente, o algoritmo **Limite** (WAHID; RAO; GOEL, 2019) estabelece um *threshold* que funciona como um critério para descarte de certos elementos. Nessa abordagem, elementos abaixo de um certo limite preestabelecido são descartados das análises. A taxa de redução obtida nessa técnica varia em função do limite configurado.

Adicionalmente, o algoritmo **Rand** (TANIZAKI; WOLFSON, 2007) se utiliza de um resultado aleatório para definir se o elemento será ou não descartado do fluxo de dados. Os níveis de redução obtidos nesta abordagem podem variar de forma significativa, uma vez que o controlador aleatório é utilizado para realizar o processo de seleção dos dados.

Conforme mencionado, existem ainda as abordagens baseadas em agrupamento. Nesse caso, as similaridades são identificadas com o fim de formar grupos ou *clusters* que possam oferecer um certo grau de representatividade frente ao conjunto de dados original. Em geral, as técnicas de agrupamento são utilizadas em contexto no qual os conjuntos de dados são estáticos.

Na presente proposta, foram utilizados os algoritmos de agrupamento **DBSCAN**, **OPTICS**, **HDBSCAN** e **X-Means**, estes algoritmos foram adaptados para garantir a sua atuação em um contexto de fluxo de dados contínuo, a fim de permitir que os conjuntos de dados recebidos do ambiente de mobilidade sejam processados conforme alcançam novas camadas do *framework* DaRkaM.

Outras propostas abordando a redução de dados podem ser encontradas na literatura, no entanto, a escolha da estratégia mais adequada passa pelo entendimento acerca do tipo de dado sob o qual se deseja implementar as estratégias de redução. Em (SADRI et al., 2022) os autores abordam diferentes trabalhos, realizando uma análise comparativa entre as diferentes estratégias de redução. Além disso, a combinação de abordagens pode ser uma alternativa para maximizar a qualidade e a capacidade de redução de dados sem prejudicar a confiabilidade ou acurácia. Neste contexto, três métricas são importantes: o tempo de processamento, a taxa de compressão e medidas de erro (SOUZA; VILLAS, 2016)(MANJORO; DHAKAR; CHAURASIA, 2016)(NADEEM et al., 2004).

Adicionalmente, cabe citar que o ambiente sob o qual se dará este processamento poderá exercer grande influência sobre a estratégia de redução mais adequada. Neste trabalho, o contexto da redução se dá no ambiente das redes veiculares. A próxima seção aborda o conceito e os aspectos gerais dessas redes, assim como os desafios ligados à comunicação, tráfego de dados e processamento nestes cenários.

## 2.3 REDES AD HOC VEICULARES

As VANET são uma subclasse da Rede *Ad Hoc* Móvel (do inglês, *MANET*), mas seu comportamento é significativamente diferente. Entre as principais diferenças está o fato de que eventuais mudanças na topologia podem impactar diretamente na operação e gerenciamento de aspectos como a comunicação e conectividade entre os elementos dispostos no espaço urbano. Analogamente, as redes VANET não possuem restrições significativas de energia, assim como não possuem potencial para operações de processamento de larga escala (YOUSEFI; MOUSAVI; FATHY, 2006).

Entre os principais benefícios das redes veiculares ou VANETs está a possibilidade de estabelecer diferentes estratégias de comunicação, onde se destacam, por exemplo, abordagens como: a comunicação *Vehicle-to-Vehicle* (V2V), na qual veículos se comunicam entre si na camada de mobilidade. Nesse contexto, cada veículo comunica-se com outros dentro do seu raio de alcance, formando uma rede *ad hoc* que opera de forma independente da infraestrutura da rodovia (GASMI; ALIOUAT, 2019; MERAIHI et al., 2010).

Adicionalmente, tem-se o formato de comunicação V2I, no qual a comunicação ocorre entre os veículos e a infraestrutura disponível ao longo das rodovias (NIARI; BERANGI; FATHY, 2018). Em geral, esta infraestrutura é composta por unidades de processamento denominadas por *Road Side Units* (RSU), estas unidades de processamento (UPs) são capazes de estabelecer comunicação com os veículos, recuperando dados como tempo, velocidade, localização entre outros.

Por fim, têm-se ainda as conexões híbridas denominadas V2X. Nesta abordagem, a comunicação pode ocorrer não só a partir da integração entre elementos V2V e V2I (VEGNI; LITTLE, 2011), como também envolvendo quaisquer outras unidades de processamento próximas à rodovia. Essa troca de mensagens baseada no conceito de *vehicle-to-everything* pode permitir a identificação de elementos situacionais, maximizando a segurança, comunicação e eficiência no trânsito. Essa proposta pode ser útil para o desenvolvimento das tecnologias para veículos autônomos. (HARDING et al., 2014).

As topologias descritas nesta seção, em especial V2V e V2I, atuam em geral sob um protocolo denominado *Wireless Access in Vehicular Environments* (WAVE). Este protocolo estabelece um padrão de conectividade para os elementos deste cenário. Este protocolo atua sob um tipo de comunicação de curta distância, estendendo seu raio por uma faixa de até 1 km, usando uma taxa de transmissão de 3 a 27Mbps (LI, 2012). Esta infraestrutura utiliza pacotes *beacons* para trocar informações entre as RSUs e os veículos. Toda transação é executada sob o protocolo IEEE 802.11p detalhado na subseção seguinte.

### 2.3.1 O Protocolo IEEE 802.11 para Redes VANET

As Redes Veiculares Ad hoc (*Vehicular Ad hoc Networks*) ou VANETs usam a comunicação sem fio em ambientes veiculares com o fim de estabelecer a conectividade entre diferentes elementos do espaço urbano. Para tanto, as redes veiculares se utilizam do protocolo *Wireless Access in Vehicular Environments* – WAVE que consiste em uma

adaptação do protocolo IEEE 802.11a para 802.11p (AL-SULTAN et al., 2014)

A definição de um padrão de comunicação em redes veiculares emerge a partir de uma iniciativa do IEEE no estabelecimento do protocolo denominado por IEEE 802.11, esta iniciativa deu origem ao amplamente utilizado protocolo 802.11p WAVE. O modelo de comunicação WAVE denota a arquitetura pela qual são padronizadas as comunicações veiculares, documentadas pelos grupos IEEE 1609 (IEEE, 2019) e IEEE 802.11p (IEEE, 2010).

O protocolo IEEE 802.11p se baseia, principalmente, no IEEE 802.11a que consiste em um padrão para redes sem fio baseado na técnica *Orthogonal Frequency-Division Multiplexing* (OFDM) ou Multiplexação por Divisão de Frequência Ortogonal. Essa técnica preconiza a compressão de *bits* contendo os dados em ondas de rádio, utilizando-se de técnicas de processamento de sinal. O protocolo 802.11a atua na taxa de frequência de 5GHz, com canais de largura de 20MHz e alcançando velocidades de até 54Mbps, o que o torna mais útil em ambientes fechados, com pouca mobilidade, em função do seu curto alcance (IEEE, 1999).

Já o protocolo 802.11p, apesar de utilizar uma abordagem similar baseada em OFDM, se difere do seu predecessor principalmente com relação ao ambiente que será utilizado, pois o IEEE 802.11p lida diretamente com a camada física e de enlace das comunicações veiculares. Nos Estados Unidos, o IEEE 802.11p opera na faixa entre 5.850GHz e 5.925GH. Nessa faixa de frequência, os nós conseguem estabelecer comunicação com uma distância máxima de 1 km, com taxa de transmissão podendo variar entre 1 Mbps e 27 Mbps. Na abordagem proposta neste trabalho, os pacotes *beacons* são transmitidos, periodicamente, através do canal de controle CCH do IEEE 802.11p. Estes canais possuem 29 bytes, mas podem exceder esse valor dependendo da personalização do projeto (RENDA et al., 2016). A Tabela 2.1 exemplifica os elementos de uma mensagem *beacon*.

Tabela 2.1: Pacote Beacon

ID Transmissor	Tamanho Tabulação	ID Veículo	ID Pacote	Latitude	Longitude	Velocidade	Cabeçalho	Tempo	...
1B	1B	1B	2B	4B	4B	4B	4B	8B	

Diferente do 802.11a, o 802.11p tem a faixa de frequência dividida em canais de 10MHz cada, sendo um canal dedicado exclusivamente para controle de comunicações - Canal de Controle (do inglês, *Control Channel* (CCH) e os demais canais utilizados para diferentes categorias de serviços de rede - Canais de Serviço ou *Service Channels* (SCH) (ALVES et al., 2009). Base para as comunicações de curto alcance dedicadas ou *Dedicated Short-Range Communication* (DSRC), o protocolo 802.11p visa estabelecer conexões mais rápidas entre os dispositivos da rede. Além das características já citadas, esse protocolo garante a transmissão dos dados entre nós que se movem com velocidade de até 200 km/h. Contudo, esse é um protocolo que possui particularidades que são essenciais para o funcionamento das *VANETs* (ALVES et al., 2008). A seção seguinte aborda a frequência de transmissão de *beacons* utilizando o protocolo 802.11p sob redes veiculares *Ad Hoc*.

### 2.3.2 Frequência de Transmissão

A frequência de transmissão dos *beacons* garante a produção contínua de dados, formando um fluxo que pode ser traduzido em uma série temporal. Uma série temporal é definida como uma coleção de observações, ordenada cronologicamente, de modo que, dependendo da aplicação e/ou cenário com alta densidade veicular, o aumento na taxa de transmissão de *beacons* pode levar o canal de controle a uma sobrecarga. Quando o canal de controle da rede é sobrecarregado, pacotes podem ser perdidos por colisão e a latência tende a aumentar, comprometendo a qualidade do serviço do sistema veicular (SCHMIDT et al., 2010) (NOORI; OLYAEI, 2013) (CHATRAPATHI et al., 2015) (SOUZA et al., 2016). A Figura 2.2 ilustra este comportamento.

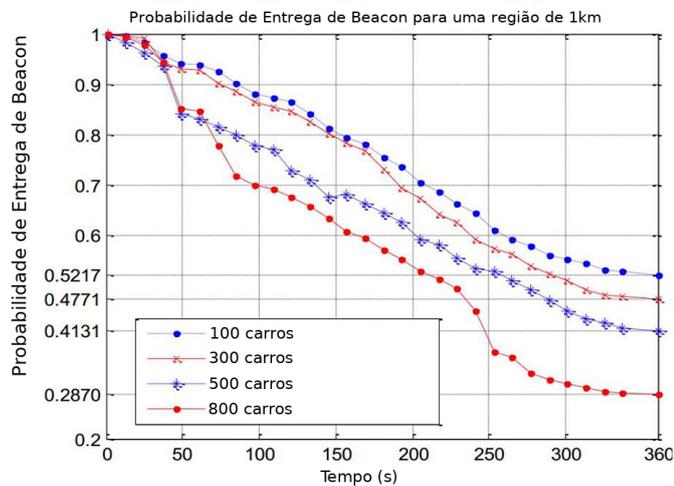


Figura 2.2: Probabilidade de entrega de beacon *vs* densidade (NOORI; OLYAEI, 2013).

Analisando a Figura 2.2, pode-se observar que a capacidade de entrega dos *beacons* é reduzida de forma mais acentuada. Conforme a densidade veicular aumenta, o canal de controle torna-se sobrecarregado, incorrendo na perda iminente de pacotes. Pode-se observar que em um cenário com 800 veículos, a probabilidade de entrega em  $\Delta_t = 360$  é de 0,2870, uma redução na ordem de 44,98% da probabilidade, se comparado a um cenário com 100 veículos.

Um desafio adicional desta pesquisa está na estruturação de um ambiente capaz de permitir a realização de testes e experimentos controlados capazes de garantir não só a reprodução de situações próximas à realidade urbana como as questões de comunicação e conectividade mencionadas nessa seção, nesse contexto os simuladores podem oferecer uma importante contribuição. A presente proposta de pesquisa se utilizou de um conjunto de simuladores que permitiram a realização dos experimentos e achados deste trabalho. Estes simuladores estão descritos na seção seguinte.

## 2.4 SIMULADORES DE TRÁFEGO E CONECTIVIDADE

Uma complexidade adicional observada no desenvolvimento de sistemas de transporte inteligente consiste na necessidade de criar ambientes capazes de emular o comportamento

dos objetos em um certo cenário urbano. Em virtude da complexidade destes sistemas em termos de infraestrutura, processamento, hardware e software, torna-se imprescindível a escolha de estratégias alternativas, capazes de permitir a realização de experimentos em ambientes controlados antes de avançar para ambientes reais. Neste contexto, o uso de simuladores torna-se uma solução eficiente para projetar ambientes, situações e desafios que podem ser analisados a partir de uma pesquisa de natureza experimental.

Os experimentos realizados com base na execução e resultados das simulações são essenciais para compor uma visão geral acerca das tecnologias, abordagens ou implantações técnicas que se deseja validar. Os resultados destes estudos retroalimentam a pesquisa a partir das observações e melhorias implementadas. Em termos gerais, as simulações possibilitam a análise prévia de uma certa abordagem por meio da experimentação e observação dos resultados (BENABDALLAH; HAMZA; BECHRIF, 2017).

Embora o uso de simuladores seja um aspecto relevante da pesquisa experimental e forneça o apoio necessário para replicar situações do mundo real em ambiente controlado, a escolha destas ferramentas pode não ser trivial. Nesse contexto, foi descrito um conjunto de critérios relevantes para esta pesquisa com base no trabalho de (CAETANO; SOARES, 2016). Em seu trabalho, os autores realizam uma avaliação detalhada do processo de seleção de simuladores:

1. O simulador tem documentação acessível;
2. O simulador dá suporte à simulação de redes veiculares;
3. O simulador possibilita a modelagem de cenários rodoviários;
4. O simulador possui licença aberta;
5. O simulador tem sido referenciado em produções acadêmicas.

Diante dos critérios apresentados por (CAETANO; SOARES, 2016), aliado a análise dos desafios dessa pesquisa, foram definidos três simuladores a serem utilizados, são eles: (SUMO, 2019), (OMNET++, 2019) e (VEINS, 2019). Cabe citar, que este conjunto de simuladores tem ganho significativa relevância, sendo utilizado em diferentes pesquisas, especialmente em trabalhos que envolvem o uso de redes VANET (CAETANO; SOARES, 2016). Observa-se ainda que este conjunto de simuladores opera de forma harmônica para construir os cenários de mobilidade capazes de atender não só os requisitos de tráfego, mas também aspectos de comunicação e conectividade, necessários nessa pesquisa. A Figura 2.3 ilustra uma arquitetura conceitual contendo os simuladores escolhidos e como estes se integram.

Analisando o papel do simulador (SUMO, 2019) (Figura 2.4a), sua configuração permite simular o tráfego rodoviário contínuo a partir de uma estrutura de rodovias, faixas, veículos e outros elementos encontrados no espaço urbano. Trata-se de um pacote de simulação altamente versátil, capaz de projetar cenários reais a partir da construção de mapas ou integrações com a ferramenta *OpenStreetMap*<sup>1</sup>. Além disso, esta solução permite realizar o controle dos elementos da rodovia, garantindo a execução e replicação de

---

<sup>1</sup><https://www.openstreetmap.org/>

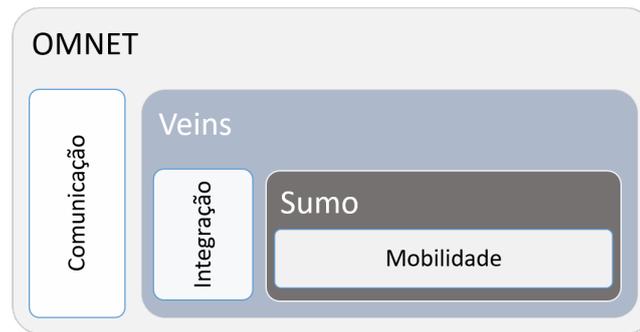


Figura 2.3: Arquitetura Integrada dos Simuladores

experimentos que podem contribuir para aferição de valores como: velocidade, distância, uso de combustível, eliminação de  $CO_2$ , entre outros métodos e técnicas disponíveis nessa solução. A Figura 2.4 ilustra o ambiente, os simuladores SUMO e sua integração com o OMNET++, utilizados nessa pesquisa.

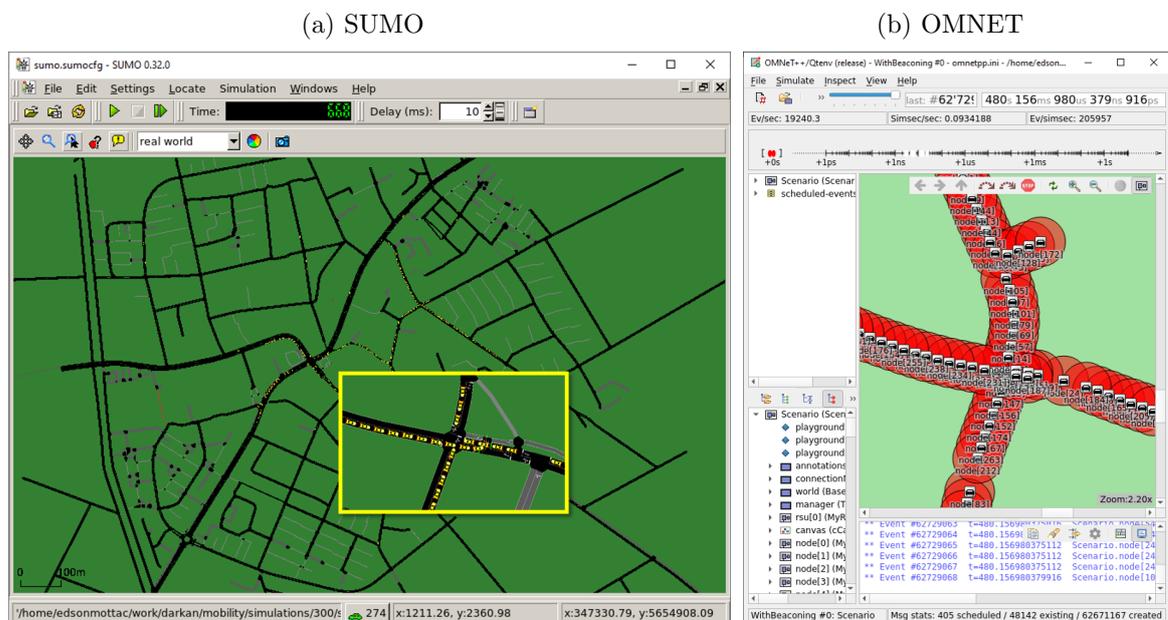


Figura 2.4: Visão dos Simuladores (SUMO, 2019) e (OMNET++, 2019) em execução

No que se refere ao (OMNET++, 2019), este consiste em um ambiente de simulação baseado em C++. Sua arquitetura conta com um conjunto de *frameworks* que juntos fornecem uma série de funcionalidades relativas à simulação de redes e conectividade. A solução oferece suporte à construção de redes de sensores, redes *Ad Hoc* sem fio e protocolos de internet. Assim, esta ferramenta é responsável por viabilizar os aspectos de conectividade do ambiente de mobilidade. No contexto deste trabalho, o OMNET++ foi utilizado para simular o padrão de conectividade IEEE802.11p, sob o qual *beacons* são emitidos e trafegados por meio dessa rede entre os elementos que compõem o framework proposto nesta pesquisa.

Por fim, para estabelecer a comunicação entre os ambientes de simulação SUMO e

OMNET, tornou-se necessário o uso de um terceiro simulador. O (VEINS, 2019) consiste em um *framework* desenvolvido para executar simulações em redes veiculares. O papel do simulador VEINS, no contexto deste trabalho, é integrar as definições de mobilidade gerenciadas pelo simulador SUMO com os aspectos de conectividade gerenciados pelo OMNET++. Esta integração promove uma rede veicular ad hoc sob a qual os experimentos propostos neste trabalho foram conduzidos.

Diante disso, pode-se observar que os simuladores apoiam a estruturação de um ambiente capaz de emular os desafios de mobilidade, garantindo a realização de experimentos preparados para avaliar não só as estratégias de redução de dados, foco desta pesquisa, mas também a capacidade das abordagens propostas em identificar as condições de tráfego em um dado instante de tempo. Assim, conforme novos dados são gerados a partir dessa infraestrutura simulada, uma unidade de processamento deve ser capaz de receber, processar e analisar, de forma contínua, estes conjuntos de dados com o fim de identificar as variações de densidade ocorridas durante o curso da simulação. Para categorizar essas variações foi utilizada a métrica LOS, descrita na seção seguinte.

## 2.5 CLASSIFICAÇÃO DO TRÁFEGO POR NÍVEL DE SERVIÇO - LOS

O *Highway Capacity Manual* (HCM) foi criado pelo *United States Transportation Research Board* (BOARD, 2010) na forma de equações matemáticas como um meio para estimar o Nível de Serviço (LOS, acrônimo para *Level of Service*, em inglês) de tráfego. LOS atua como um classificador no qual seis diferentes níveis de serviço são definidos, com o fim de fornecer uma referência consistente acerca das medições usadas para descrever as condições do fluxo de tráfego em uma certa rodovia.

Cada um dos níveis definidos no nível de Serviço (LOS) representa valores relativos à densidade veicular identificada nas rodovias, estes níveis são classificados com base em elementos como a velocidade média dos veículos e a velocidade da rodovia. Assim, para descobrir a condição de um fluxo de tráfego dado por um grau LOS, um peso ( $w$ ) representado por  $w = \{w_1, w_2, \dots, w_i\}$  é calculado para cada borda ( $e_i$ ) com uma equação que considera a densidade do tráfego rodoviário e a velocidade média deste mesmo tráfego.

Os pesos  $w_i$ , contidos na equação, terão como valor resultante um número decimal inversamente proporcional às condições de tráfego, ou seja, estradas de fluxo livre têm menores pesos do que estradas congestionadas, além disso, a taxa de utilização da estrada está associada ao peso, variando de  $w = 0$  até  $w = 1$ . Assim, a condição de tráfego é dada em cada estrada  $e_i$  com base nestes respectivos pesos  $w_i$ . Esta relação é mostrada na Equação 2.1.

$$w_i = 1 - \frac{v_i^{avg}}{v_i^{max} \times d_i} \mid d_i > 0 \quad (2.1)$$

Onde  $v_i^{avg}$ ,  $v_i^{max}$  e  $d_i$  representam a velocidade média, velocidade máxima permitida na estrada e densidade, respectivamente. O peso  $w_i$ , obtido a partir do valor resultante da Equação 2.1, consiste em um valor normalizado entre  $[0..1]$ , sob o qual será realizada a classificação por nível de serviço. O processo de classificação consiste na segmentação

deste intervalo definindo faixas que variam progressivamente de [A], que representa o trânsito livre, até a classe [F], que indica um fluxo veicular totalmente congestionado. A Figura 2.5 ilustra a relação entre a velocidade e a densidade veicular, ilustrando ainda como este comportamento é refletido na classificação por nível de serviço (LOS).

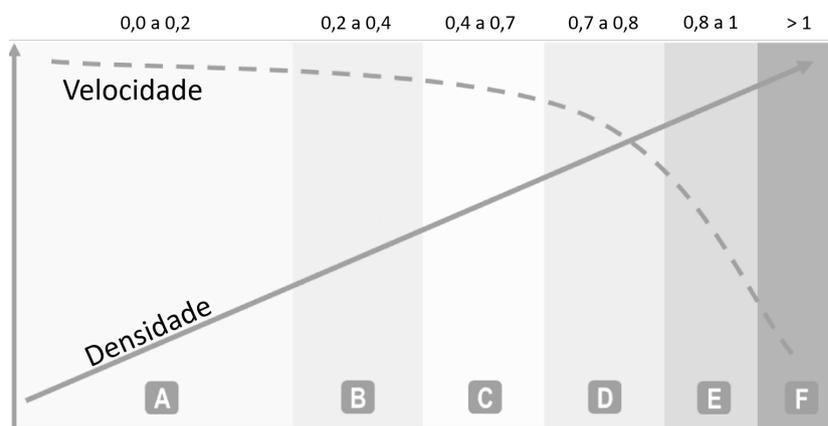


Figura 2.5: Relação entre a taxa de velocidade x capacidade da rodovia (BOARD, 1997)

Analisando a Figura 2.5, pode-se observar que conforme a densidade veicular aumenta, a velocidade é concomitantemente reduzida, assim, a classificação por nível de serviço preconiza a identificação desses diferentes estágios de um congestionamento. Os níveis de serviço utilizados nesse método e a classificação são descritos mais detalhadamente, a seguir:

- *Free flow traffic*: [A] - Tráfego Livre: Nesse intervalo, os usuários da rodovia praticamente não são afetados pela presença de outros veículos em um trecho da estrada. A escolha da velocidade e a capacidade de manobra são livres. Existe um alto nível de conforto, uma vez que é requerida menor atenção do motorista durante a condução dos veículos.
- *Reasonably free-flow*: [B] - Tráfego Razoavelmente Livre: São sentidos os efeitos relativos à presença de outros veículos e isso, por sua vez começa a afetar o comportamento dos motoristas, individualmente. Adicionalmente, sua capacidade de manobra torna-se reduzida. Nesse intervalo de classificação LOS, assume-se que o conforto é mantido, mas já requer algum esforço para manter-se atento no que se refere a eventuais veículos próximos.
- *Stable-flow*: [C] - Tráfego Estável: Nessa faixa de classificação, a presença de outros veículos afeta os motoristas. A escolha da velocidade é afetada pelo aumento da densidade veicular e a capacidade de manobra requer atenção. O nível de conforto diminui, uma vez que a percepção de proximidade de outros veículos torna-se mais intensa.
- *Approaching unstable-flow*: [D] - Próximo à Instabilidade: Tanto a escolha da velocidade quando a capacidade de manobra do condutor são severamente reduzidas.

Nessa faixa de classificação tem-se baixo nível de conforto, pois as colisões com outros veículos devem ser constantemente evitadas.

- *Unstable-flow*: [E] - Tráfego Instável. Velocidade baixa, mas uniforme. A capacidade de manobra só é possível sob restrição de outro veículo. Nesse estágio, o usuário do sistema tende a frustrar-se, reduzindo de forma significativa o conforto na condução do veículo;
- *Breakdown-flow*: [F] - Tráfego Congestionado. Velocidade instável com formação de filas de espera em vários pontos das rodovias. Essa classe é caracterizada pelos ciclos de paradas e partidas sem padrão aparente que são influenciados pelo comportamento dos outros motoristas. Nessa condição, tornam-se necessários altos níveis de vigilância e não há conforto na condução. Nesse nível, a relação volume/-capacidade excede 1, o que implica que o segmento de estrada é usado acima da capacidade para a qual foi projetada.

Observa-se, portanto, que os níveis de serviço são classificados com base nas condições da rodovia. Neste trabalho, a métrica LOS foi utilizada para atuar na identificação de congestionamento por meio do acompanhamento contínuo das eventuais mudanças na densidade veicular. Para isso, um desafio adicional deste trabalho está relacionado à análise e classificação dos níveis de serviço utilizando uma abordagem baseada em fluxo de dados contínuo. Essa abordagem preconiza o acompanhamento do estado do tráfego de modo a contribuir com a identificação e distribuição de novas rotas, com o fim de manter a eficiência do tráfego. A seção seguinte apresenta desafios e soluções em uma abordagem baseada em fluxo contínuo de dados.

## 2.6 FLUXO DE DADOS CONTÍNUOS

Os avanços tecnológicos das últimas décadas, em termos de conectividade, têm impactado de forma significativa os meios tradicionais pelos quais são realizadas ações como: seleção, análise e armazenamento de dados. Atividades como a mineração e aprendizado de máquinas tornam-se cada vez mais complexas, na medida em que os conjuntos de dados tornam-se maiores e sua geração passa a operar de forma gradual e contínua. Nesse cenário, pesquisas têm sido desenvolvidas com o fim de estabelecer um novo paradigma capaz de modificar a dinâmica do processo de análise de dados, deixando de operar sobre conjuntos limitados e estáticos e passando a operar sob a forma de fluxos contínuos e potencialmente ilimitados. A este fluxo contínuo de dados convencionou-se chamar de *Data Streams* ou Fluxo Contínuo de Dados (FCD) (GAMA; (EDS), 2007)

Assim, diferentemente dos conjuntos de dados estáticos, sob os quais as análises são realizadas com base na completude dos dados disponíveis, em um ambiente de fluxo de dados contínuo, este conhecimento precisa ser construído no decurso das análises. Conforme novas entradas são processadas, novos conhecimentos acerca do comportamento dos dados podem ser revelados (PEIXOTO et al., 2021).

Pode-se definir um fluxo de dados  $S$  como uma sequência ininterrupta de elementos  $x_1, x_2, \dots, x_N$  de modo que  $S = \{x^i\}_{i=1}^N$ , onde  $N$  tende potencialmente ao infinito ( $N \rightarrow$

$\infty$ ) (SILVA et al., 2013). No entanto, observa-se que a análise proposta em (SILVA et al., 2013) abarca em sua estrutura apenas um fluxo de dados unidimensional, ou seja, sua estrutura permite que seja trafegada uma informação a cada interação. Sensores como unidade ou temperatura são um exemplo deste tipo de fluxo de dados.

Entretanto, com o avanço da tecnologia e a necessidade de estabelecer fluxos de dados mais eficientes, muitos autores passaram a debruçar-se sobre o conceito de fluxo de dados contínuo e multidimensional (SOUSA et al., 2006; NUNES et al., 2013; SILVA et al., 2016; CYGANNEK, 2018). Um fluxo de dados multidimensional pode ser descrito como um conjunto de elementos  $X_i$  e um fluxo  $DS$  representado por um vetor de atributos  $n$ -dimensional  $x_i = [x_j^i]_{j=1}^n$  pertencendo a um espaço de atributos que pode ser contínuo, categórico ou ambos. Nesse contexto, um único fluxo de dado pode gerar múltiplas informações (LIMA, 2022). Entre os domínios nos quais uma abordagem baseada em fluxo de dados contínuo pode ser aplicada estão (SPINOSA, 2015):

- **Redes de sensores:** Sensores interconectados que podem fornecer dados de forma contínua acerca de um ou mais elementos sensoreados.
- **Serviços web:** Sites ou portais capazes de acompanhar a navegação de grandes grupos de usuários, a fim de monitorar a sua experiência, desempenho e ações na plataforma.
- **Mercado financeiro :** Monitoramento contínuo de variações na bolsa de valores, análise de dados financeiros em tempo real.
- **Redes de computadores:** Análise do desempenho das redes de computadores, a fim de identificar variações que justifiquem a aplicação de ajustes, configurações de banda, balanceamentos ou ainda a identificação do comportamento da rede para prevenção de ataques.

Adicionalmente, o modelo de fluxo de dados contínuo se caracteriza, essencialmente, pelos seguintes pontos (BABCOCK et al., 2002):

- Os elementos que compõem os dados são recebidos de maneira contínua;
- O sistema não tem controle sobre a ordem em que os elementos são recebidos;
- Fluxos de dados contínuos são potencialmente ilimitados.
- Uma vez que um elemento do fluxo tenha sido processado, deve ser descartado ou arquivado, não podendo ser recuperado facilmente, exceto se for explicitamente armazenado na memória, que é pequena quando comparada ao fluxo.

Cabe destacar ainda que o uso de algoritmos de mineração, aprendizado de máquina, sejam estes relacionados à formação de agrupamento ou classificação de dados, requerem um conjunto de dados representativo. No entanto, em um ambiente de fluxo contínuo de dados, essa representatividade pode ser perdida, pois a cada interação, novos fluxos de dados são recebidos para processamento e este processo persiste enquanto existirem

dados a serem processados, esta dinâmica é ilustrada no trabalho de (PEIXOTO et al., 2020). Para lidar com este comportamento, uma alternativa pode estar no uso de janelas temporais. A Figura 2.6 ilustra esse funcionamento.

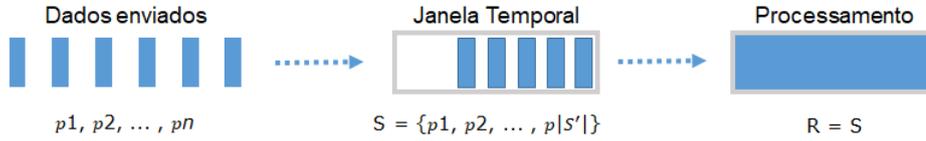


Figura 2.6: Representação conceitual do funcionamento das Janelas Temporais

Analisando a Figura 2.6 pode-se observar que a janela temporal está representada por um conjunto  $S$ , capaz de receber subconjuntos de elementos particionados em uma série de segmentos  $S' = \{p_1, p_2, \dots, p_{|S'|}\}$ , tão logo o tamanho da janela é alcançado  $\gamma$ , os resultados são consolidados para futuro processamento. Assim, pode-se observar que as janelas temporais contam com um mecanismo para ajustar limites em um fluxo ilimitado de dados. Tais janelas garantem a criação de um *buffer* capaz de reunir os dados por um certo tempo, liberando-os em pequenos lotes para processamento tão logo um limite de tempo seja alcançado. Essa dinâmica na movimentação dos dados garante que o processamento ocorrerá sobre um certo conjunto de dados que representam, em última análise, os dados mais atuais dos diferentes fluxos de dados recebidos (TOLPIN, 2016) e (PATROUMPAS; SELLIS, 2006).

Embora estas alternativas tenham se mostrado eficientes no que se refere ao processamento dos dados com o fim de evidenciar um certo cenário atual, a realização de previsões sobre estes conjuntos ainda é uma tarefa desafiadora. Algoritmos de predição requerem um entendimento prévio sobre o conjunto de dados que são utilizados na formação de um grupo de treinamento. Em um ambiente de fluxo de dados contínuo, o estabelecimento de um conjunto estático de treinamento pode levar a invariável perda de acurácia, uma vez que, conforme novos dados são recuperados, este comportamento descrito no conjunto de treinamento torna-se desatualizado. Este conceito é conhecido como *Concept Drift*.

Para mitigar esse comportamento, algoritmos de detecção de mudança de conceitos podem ser implementados. Tais algoritmos procuram identificar a ocorrência destas variações, garantindo que um certo conjunto de treinamento esteja adequado às últimas ocorrências de um fluxo de dados. Abordagens capazes de lidar com o fenômeno do *Concept Drift* podem ser vistas em (ANGEL; BAROLO; ERNESTINA, 2016), (JĘDRZEJOWICZ; JĘDRZEJOWICZ, 2015) e (KRAWCZYK; WOŹNIAK, 2015), apenas para citar algumas, no entanto, não foram localizadas abordagens voltadas à utilização de desvios de conceito no campo das redes veiculares ou na computação urbana.

## 2.7 ESTRATÉGIAS DE AGRUPAMENTOS BASEADOS EM DENSIDADE

Redes veiculares são caracterizadas pelo fluxo contínuo de dados trafegados. Veículos podem emitir mensagens que eventualmente podem ser capturadas pela infraestrutura ou mesmo por outros veículos em um processo contínuo que se repete em intervalos que, em geral, podem ser medidos em milissegundos. Neste contexto, as técnicas de agrupamento

podem ser utilizadas para encontrar similaridades baseadas em características chave que permitam a identificação de elementos mais representativos (MOHANTY et al., 2022).

Em linhas gerais, os algoritmos de agrupamento permitem a conversão de conjunto de dados heterogêneos em pequenos grupos homogêneos. Esse processo ocorre por meio da utilização de medidas de similaridade. Essas métricas permitem a análise de elementos com características em comum e utilizam as semelhanças para formar os agrupamentos (*clusters*, em inglês). Os grupos criados a partir da análise da similaridade formam um conjunto representativo do conjunto de dados original. Por outro lado, tende-se a acreditar que dados que não possuem pares em termos de similaridade, tenham menor relevância nesses mesmos conjuntos. Tal hipótese foi validada ao longo deste trabalho. Os agrupamentos derivados da análise das similaridades podem ainda ser utilizados para tarefas como a análise de padrões, formação de subgrupos de dados, auxiliar na tomada de decisões, além do suporte ao aprendizado de máquina, incluindo mineração de dados, recuperação de documentos, segmentação de imagens e classificação de padrões.

Um recurso utilizado pelas técnicas de agrupamento consiste em criar micro-grupos para armazenar as estatísticas dos grupos. Um micro-grupo pode ser observado como uma estrutura  $(n, SL, SQ)$ , em que:  $n$  representa o número de elementos de um grupo,  $SL$  a soma linear dos elementos e  $SQ$  a soma do quadrado. As informações contidas em um micro-grupo são suficientes para calcular algumas medidas estatísticas. Micro-grupos têm as propriedades de incremento e aditividade, a propriedade de incremento é usada quando um novo elemento é adicionado a um grupo e a de aditividade é usada quando dois grupos se juntam. A presente pesquisa se utiliza de diferentes técnicas baseadas em agrupamento para detecção de congestionamento e suporte à decisão em um contexto de fluxo de dados contínuo na névoa; entre os algoritmos utilizados destacam-se *Density-Based Spatial Clustering of Applications with Noise* (DBSCAN), *Ordering points to identify the clustering structure* (OPTICS), o algoritmo X-Means, uma extensão do conhecido algoritmo de agrupamento k-Means, *Hierarchical Density-based Spatial Clustering of Applications with Noise* (HDBSCAN), que estende características do conhecido algoritmo DBSCAN para incorporar a natureza hierárquica ao processo de formação e identificação de similaridades; além de outras propostas para redução de dados na névoa baseadas em amostragem simples. A utilização de técnicas de agrupamento com o fim identificar similaridades em dados de VANET foi explorada em outros trabalhos, tais como: (MANJORO; DHAKAR; CHAURASIA, 2016), (NADEEM et al., 2004), (SOUZA; VILLAS, 2016), (Chiti; Fantacci; Rigazzi, 2014). As próximas seções descrevem os algoritmos de agrupamento utilizados nessa pesquisa.

### 2.7.1 A Técnica de Agrupamento Espacial Baseada na Densidade Aplicada com Ruído (DBSCAN)

Algoritmos de agrupamento baseados em densidade espacial têm cumprido um papel relevante no que se refere à análise de dados e descoberta de padrões e similaridades em dados espaciais, tais algoritmos empregam uma abordagem capaz de identificar grupos em formatos diversos, nos quais agrupamentos são definidos com base na densidade das regiões (EL-SONBATY; SAID, 2009)

O algoritmo de agrupamento espacial baseado na densidade aplicada com ruído – DBSCAN emerge como uma alternativa aos algoritmos de agrupamento tradicionais em função da sua capacidade de comportar três requisitos fundamentais, a saber: 1) Conhecimento do domínio para definição dos parâmetros de entrada. O conhecimento do domínio permite a definição de parâmetros básicos que podem, em certa medida, definir a eficiência do algoritmo para um certo conjunto de dados; 2) Descoberta arbitrária de grupos/*clusters*. O algoritmo é capaz de identificar, baseado na análise da amostra, o número de grupos ideal para um certo conjunto e, por fim, 3) Manter níveis de eficiência aceitáveis para grandes conjuntos de dados (ESTER et al., 1996).

Desenvolvido por Ester et al. (1996), o DBSCAN forma seus grupos baseando-se em áreas densas separadas por áreas menos densas. Para sua utilização é necessário definir, em princípio, os parâmetros  $\epsilon$ , que denotam o raio da vizinhança a partir de uma certa posição, e  $MinPts$  o qual consiste no número mínimo de vizinhos necessários para que os achados em termos de similaridade sejam caracterizados como agrupamento. Este processo pode ser melhor analisado a partir de algumas definições específicas ilustradas a seguir (CASSIANO, 2015):

**Definição 1.** *Raio de vizinhança ( $\epsilon$ ):* O elemento vizinho de um objeto  $p$ , com raio  $\epsilon$ , que pode ser chamado de  $\epsilon$ -vizinhança de  $p$  é dado por:  $N(p) = \{q \in D \mid dist(p, q) < \epsilon\}$ . A Figura 2.7 ilustra essa relação.

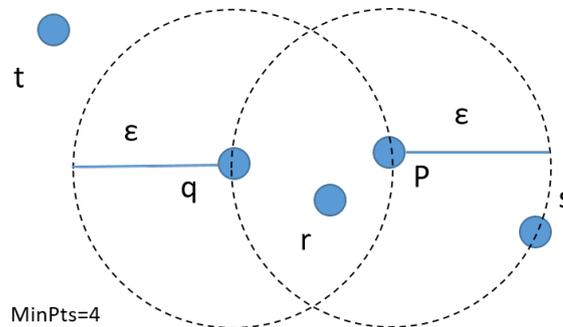


Figura 2.7: Relação do raio de vizinhança. Adaptado de (CASSIANO, 2015)

Analisando a Figura 2.7, pode-se observar que para cada ponto disponível existe pelo menos um número de objetos que atendam ao  $MinPts$ , dentro do seu raio de vizinhança, entretanto, para o algoritmo DBSCAN, deve-se considerar a existência de dois tipos consecutivos de pontos, são eles os 1) pontos centrais, contidos no interior do cluster, estes pontos representam, em linhas gerais, a densidade dos grupos formados e os pontos situados na fronteira definidos como 2) pontos de borda. Estes possuem elementos inferiores em um certo  $\epsilon$ , mas ainda assim, estão na vizinhança de um ponto central. As Definições 2 e 3 descrevem o comportamento e as características destes pontos.

**Definição 2.** *Ponto Central:* Quando o raio de vizinhança de um objeto atende a quantidade definida por  $MinPts$ , então trata-se de um ponto central. A Figura 2.7 ilustra

essa característica, nessa figura, pode-se observar, portanto, que para um  $MinPts = 4$ ,  $P$  pode ser caracterizado como um ponto central.

**Definição 3.** *Ponto Borda:* De forma contrária à Definição 2, se o raio de vizinhança contém um conjunto de pontos abaixo do  $MinPts$ , mas mantém pelo menos um ponto central, ele poderá ser classificado como um ponto de borda.

A partir das definições apresentadas, observa-se que, em geral, o raio de vizinhança de um ponto de borda conterà menos objetos do que um certo ponto central. Diante disso, a escolha de um  $MinPts$  adequado pode aumentar de forma significativa a qualidade dos *clusters* resultantes deste processamento. Adicionalmente, os objetos que não integram nenhum grupo são classificados como ruído. Para mitigar a ocorrência de ruídos, pode-se estabelecer que para cada  $P$  em um *cluster*  $C$  exista um  $Q$  em um *cluster*  $C$ , assim,  $P$  está dentro de  $\epsilon$  (raio de vizinhança) de  $Q$  e  $P$  e atendendo os requisitos definidos para  $MinPts$ . A Figura 2.8 ilustra essa condição.

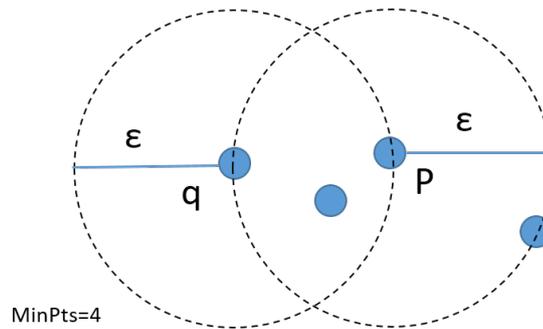


Figura 2.8: Relação do raio de vizinhança, adaptado de (CASSIANO, 2015)

Cabe observar ainda que nessas condições,  $P$  é capaz de alcançar por densidade direta de  $Q$ , pois estão dentro de um mesmo grupo, no entanto,  $Q$  não é alcançável por densidade direta de  $P$ , pois este não é um ponto central. Neste segundo caso, trata-se de uma densidade não simétrica. O alcance por densidade consiste no mecanismo pelo qual é realizado o processo de descoberta dos pontos, identificação das similaridades e formação dos agrupamentos. Estes conceitos são descritos formalmente na Definição 4.

**Definição 4.** *Alcance Direto por Densidade:* Um objeto  $p$  é alcançável por densidade diretamente do objeto  $q$ , dentro de um certo raio  $\epsilon$  e um certo valor de  $MinPts$ , se  $p$  está na  $\epsilon$  vizinhança de  $q$ , e  $q$  é um ponto central.

Pode-se observar, portanto, que o alcance direto por densidade é simétrico para pares de pontos centrais. Analogamente, quando um ponto central e um ponto de borda são envolvidos, esse alcance será chamado de assimétrico. Neste caso, pode-se concluir que  $p$  seria alcançável por densidade diretamente de  $q$ ; mas  $q$  não seria alcançável por densidade diretamente de  $p$ , porque  $q$  não é ponto central (CASSIANO, 2015). O fluxo de formação

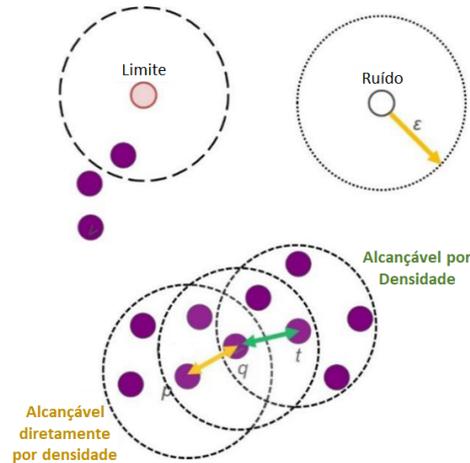


Figura 2.9: Métodos de Agrupamento DBSCAN (GIACOUMIDIS et al., 2019)

de agrupamento é ilustrado na Figura 2.9, onde são descritos os elementos utilizados e suas interações.

Analisando a Figura 2.9, pode-se observar que a vizinhança de um ponto  $p_i$  é determinada por  $dist(p_i, p_j) \leq \epsilon$ , em que  $dist(p_i, p_j)$  é a distância euclidiana entre os pontos  $p_i$  e  $p_j$ . O número de pontos que pertencem a esta vizinhança é determinado por  $n_\epsilon(p_i)$ . O DBSCAN inicia selecionando aleatoriamente um ponto  $p_i$ , se  $n_\epsilon(p_i) < MinPoints$  este ponto é um ruído. Por outro lado, se  $n_\epsilon(p_i) \geq MinPoints$  ele é chamado ponto principal, então um novo agrupamento é criado. Conforme novos pontos são observados na condição de  $n_\epsilon(p_i) < MinPoints$  e, se um de seus vizinhos for um ponto principal, este ponto que era classificado como ruído é então reclassificado como ponto de borda e um novo grupo é formado, assim o DBSCAN diferencia pontos de borda e pontos ruidosos (ALI; ASGHAR; SAJID, 2010). O processo de agrupamento termina quando todos os pontos forem visitados e nenhum novo ponto seja adicionado a nenhum dos grupos já formados.

## 2.7.2 Ordering Points to Identify the Clustering Structure (OPTICS)

Uma abordagem alternativa para a identificação de similaridades com base na formação de agrupamentos por densidade consiste no algoritmo OPTICS (ANKERST et al., 1999). Inspirado no DBSCAN, esta abordagem tenta aprimorar sua estrutura para identificar *clusters* mais significativos mesmo em densidades variáveis. Para muitos conjuntos de dados reais não é possível caracterizar sua estrutura de agrupamento utilizando um único parâmetro global de densidade, assim, o algoritmo OPTICS fundamenta sua estratégia no processamento simultâneo de variados valores de  $\epsilon$ , de modo a permitir a construção de diferentes agrupamentos, simultaneamente. A Figura 2.10 ilustra um exemplo dessas diferenças de densidades observadas entre diferentes *clusters*.

O algoritmo OPTICS atua armazenando resultados baseados na ordem de processamento dos pontos, esta estratégia procura ordenar os pontos linearmente de forma que

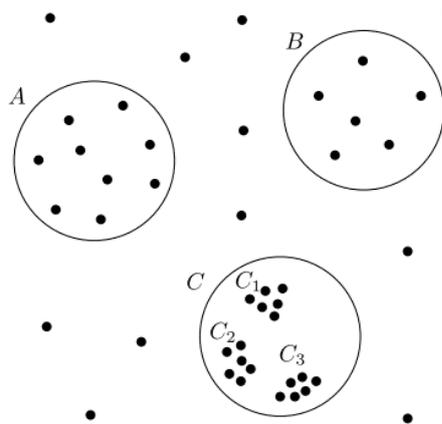


Figura 2.10: Diferentes densidades, adaptado de (ANKERST et al., 1999)

os pontos mais próximos tornam-se vizinhos de ordenação. Adicionalmente, para cada ponto são calculadas as distâncias as quais deverão representar a densidade para um certo *cluster* atendendo ao critério definido pelo  $Min_{pts}$ . A distância entre os objetos é analisada a partir de dois conceitos fundamentais, a saber:

- **a) Distância do Núcleo (Core Distance).** Que consiste no cálculo realizado com base no raio épsilon  $\epsilon$ , e procura classificar um certo ponto como central. Neste contexto, cada ponto é avaliado como referência, a fim de identificar os elementos centrais para um certo raio. Se a centralidade não é alcançada, a distância torna-se indefinida.
- **b) Distância de Acessibilidade (Reachability Distance).** Tem por função analisar a relação com outros pontos do *dataset*, assim, a distância de acessibilidade se dá entre um ponto  $p$  e  $q$ , de modo que a maior distância de  $p$  para qualquer outro ponto será dada por essa mesma distância entre  $p$  e  $q$ .

A Figura 2.11, ilustra as diferenças entre as respectivas distâncias armazenadas.

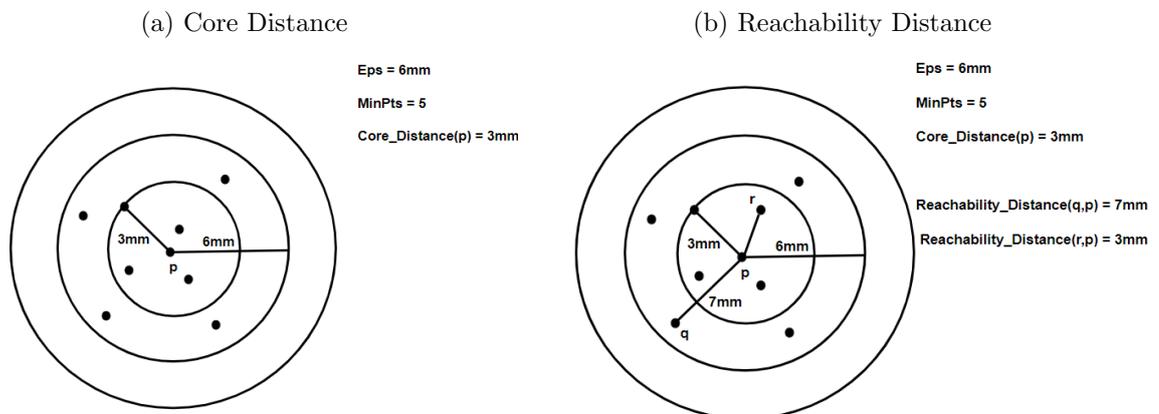


Figura 2.11: OPTICS Terminology (THILAGAMANI; JAYANTHILADEVI; ARUNKUMAR, 2018)

Assim, com base nos conceitos analisados acerca das distâncias e suas respectivas características, pode-se depreender as definições, a seguir. A Figura 2.12 ilustra as definições.

**Definição 5.** *Core Distance:* Seja  $p \in D$ , seja  $\epsilon$  um valor de distância de  $p$ , seja  $p_\epsilon N$  a vizinhança de  $p$  e  $MinPts$  um número natural. Então, a *core distance* é definida como:

$$core\_distance_{\epsilon, MinPts(P)} = \begin{cases} \text{Não definida, se } Cardinality(N_\epsilon(p)) < MinPts \\ MinPts\_distance(p), \text{ caso contrário} \end{cases}$$

Analogamente, a distância de acessibilidade (*reachability distance*) pode ser depreendida a partir da seguinte definição:

**Definição 6.** *reachability distance:* sejam  $p \in D$  e  $o \in D$ . A *reachability distance* de  $p \rightarrow o$  pode ser definida como:

$$reachability\_distance_{\epsilon, MinPts(p,o)} = \begin{cases} \text{Não definida, se } |N_\epsilon(o)| < MinPts \\ \max(core\_distance(o), distance(o, p)), \text{ caso contrário} \end{cases}$$

A Figura 2.12 ilustra uma visualização contendo os conceitos apresentados.

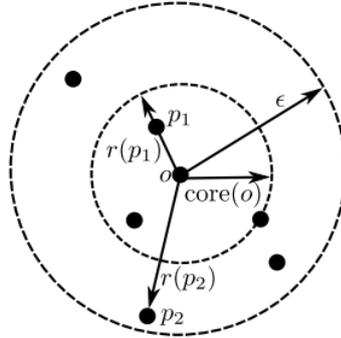


Figura 2.12: Representação das definições para um  $MinPts = 4$ . (ANKERST et al., 1999)

Com base na aferição das distâncias, um conjunto de metadados é reunido contendo não só a ordem de processamento dos objetos, mas também suas respectivas distâncias calculadas para cada objeto. Nesse contexto, os objetos são selecionados de forma que sejam alcançáveis por densidade com o menor valor de  $\epsilon$ , deste modo, pode-se garantir que grupos de alta densidade sejam processados primeiro. Esse processo ocorre a partir da extensão dos conceitos preconizados pelo algoritmo DBSCAN.

A estrutura original do algoritmo DBSCAN preconiza que para cada *Core Object* de um grupo, sua vizinhança em um dado raio  $\epsilon$  atenda ao critério definido pelo  $MinPts$  de objetos. Essa estrutura pode ser descrita como:  $N_\epsilon(p) = \{q \in D \mid dist(p, q) \leq \epsilon\}$ . Nesse contexto, um objeto diretamente alcançável por densidade pode ser descrito conforme a seguinte definição: (ANKERST et al., 1999)

**Definição 7.** O objeto  $p \in D$  é diretamente alcançável por densidade a partir de  $q \in D$  se:

1.  $p \in N_{\epsilon}(q)$
2.  $Cardinality(N_{\epsilon}(q)) \geq Min_{pts} \rightarrow foreach(coreobject)$

Entretanto, cabe observar que tal associação implica que só será possível para um certo objeto ser diretamente alcançável por densidade se o ponto de partida for um *core object*.

Analogamente, um objeto é alcançável por densidade quando alinha-se a seguinte definição:

**Definição 8.** O objeto  $p \in D$  é alcançável por densidade a partir de  $q \in D$  se existe uma cadeia de objetos  $p_1, \dots, p_n$ ,  $p_1 = q$  e  $p_n = p$  se  $p_i + 1$  é diretamente alcançável por densidade a partir de  $p_i$

Além disso, diz-se que um objeto é conectado por densidade quando:

**Definição 9.** O objeto  $p \in D$  é conectado por densidade a  $q \in D$  se existe um objeto  $o \in D$  tal que ambos  $p$  e  $q$  são alcançáveis por densidade a partir de  $o$ .

Essas definições são ilustradas na Figura 2.13, considerando um  $Min_{pts} = 3$ .

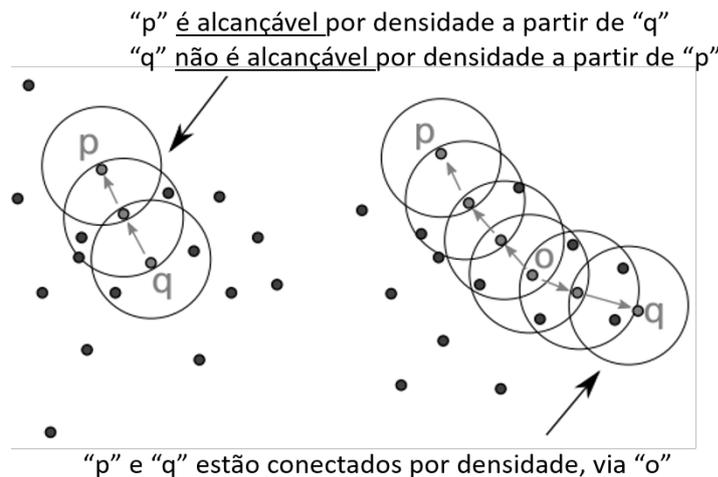


Figura 2.13: Medidas de conectividade. Adaptado de (ANKERST et al., 1999)

Por fim, (ANKERST et al., 1999) argumenta que um *cluster*  $C$  é um subconjunto não vazio de  $D$  que satisfaz as seguintes condições:

**Definição 10.** O objeto  $p \in D$  é diretamente alcançável por densidade a partir de  $q \in D$  se:

1. *Maximalidade:*  $\forall p, q \in D$  : se  $p \in C$  e  $q$  é alcançável por densidade a partir de  $p$ , então  $q \in C$
2. *Conectividade:*  $\forall p, q \in C$  :  $p$  é conectado por densidade a  $q$

Analisando as definições frente a outros algoritmos baseados em agrupamento convencionais, pode-se observar que o algoritmo OPTICS utiliza uma abordagem ligeiramente diferente, uma vez que a proposta de (ANKERST et al., 1999) não concentra o esforço na segmentação explícita dos grupos. Em vez disso, sua proposta centra-se na construção dos metadados baseados na visualização das distâncias de acessibilidade e a utilização dessas para gerar os agrupamentos de dados. Como resultado, têm-se uma ordenação do conjunto de dados que permite extrair agrupamentos baseados em densidade para infinitas configurações de parâmetros a um custo computacional significativamente reduzido.

### 2.7.3 Hierarchical Density-based Spatial Clustering of Applications with Noise (HDBSCAN)

Os algoritmos de agrupamento baseados em densidade têm cumprido um relevante papel na qualidade da formação de *clusters*, adicionando às suas estruturas mecanismos capazes de considerar não só aspectos relativos à similaridade dos elementos, mas também seu posicionamento em um certo cenário. Neste contexto, algoritmos como OPTICS e DBSCAN, descritos nas seções anteriores, têm oferecido sua relevante parcela de contribuição.

Alternativamente, uma outra relevante abordagem é proposta por (CAMPELLO; MOULAVI; SANDER, 2013), trata-se do algoritmo *Hierarchical Density-based Spatial Clustering of Applications with Noise* ou HDBSCAN, uma extensão do algoritmo DBSCAN, cuja abordagem tem por objetivo converter os agrupamentos dando-lhes uma dimensão hierárquica a partir da qual pode-se construir uma árvore simplificada de *clusters*. A abordagem objetiva obter uma partição coesa que abarque apenas os mais significantes elementos que são selecionados com base na sua medida de estabilidade de *clusters*.

Assim como o OPTICS, a abordagem do HDBSCAN procura mitigar os desafios do DBSCAN para identificação dos agrupamentos em conjuntos de dados com densidades variadas, conforme mencionado anteriormente. Nessa abordagem, um parâmetro de densidade  $\epsilon$  é definido de forma global para todos os objetos do conjunto de dados analisado, assim, a definição de um  $\epsilon$  baixo, pode levar a grupos com menor densidade e maior formação de ruídos, por outro lado, valores muito altos podem propagar a fusão de diferentes grupos, resultando em baixa qualidade de classificação.

Para mitigar este efeito, o algoritmo HDBSCAN propõe a criação de uma árvore hierárquica que, em última análise, seria equivalente a diferentes resultados obtidos pelo DBSCAN utilizando diferentes raios de vizinhança. Entretanto, ao invés de realizar

uma métrica baseada no número de objetos dentro de um certo raio, HDBSCAN propõe uma análise baseada na distância de um ponto para seus  $K$  vizinhos mais próximos. A distância do núcleo ou (*Core Distance*) consiste na distância de um certo ponto para os seus  $K$  vizinhos mais próximos. A Figura 2.14 ilustra as densidades por regiões e seus respectivos raios.

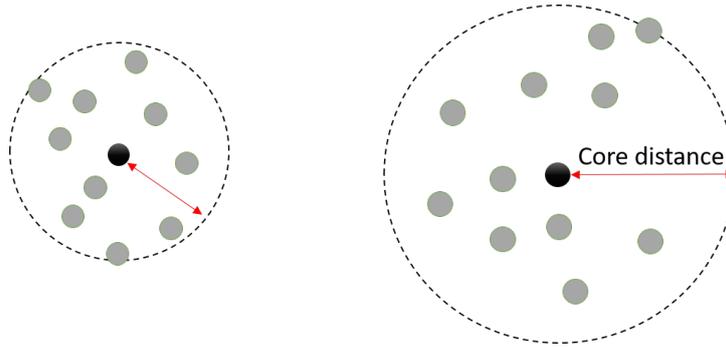


Figura 2.14:  $K = 8$ . As regiões de maior densidade (esquerda) têm um raio menor, enquanto as regiões de menor densidade têm um raio maior. Adaptado de (FILIPPE; MOURA, 2021)

Diante disso, dado um valor mínimo de pontos  $\text{Min}_{Pts}$ , o algoritmo HDBSCAN será capaz de calcular, com base nas distâncias identificadas, uma hierarquia de partição para todos os valores possíveis de  $\epsilon \in [0, \infty]$  de modo que cada nível hierárquico ou partição esteja associado a um valor de  $\epsilon$  (GERTRUDES, 2019). Cada valor é calculado com base na distância central, definida para um certo parâmetro  $k$  para um ponto  $x$ . Para distribuir os elementos com menor densidade, a abordagem utiliza a distância de acessibilidade mútua. Essa distância é definida na Equação 2.2 (MCINNES; J; S., 2016)

$$d_{mreach-k}(a, b) = \max\{core_k(a), core_k(b), d(a, b)\} \quad (2.2)$$

Onde  $d(a, b)$  é a distância métrica original entre  $a$  e  $b$ . Assim, pontos de maior valor mantêm a mesma distância entre eles e pontos esparsos são afastados a uma distância máxima, utilizando como referência a distância para o núcleo. Este processamento inicial é submetido a geração de uma árvore que dará origem a estrutura hierárquica dos agrupamentos.

Com base nessa abordagem, o algoritmo HDBSCAN é capaz de construir um grafo unidirecional, baseado no conceito de árvore geradora mínima (*minimum spanning tree* - MST), que representa todos os pontos de dados identificados em um certo conjunto de dados. Nesse grafo, cada vértice representa um ponto e as arestas um valor ponderado da distância de acessibilidade mútua entre um par de pontos (MCINNES; J; S., 2016). A Figura 2.15 ilustra o grafo resultante.

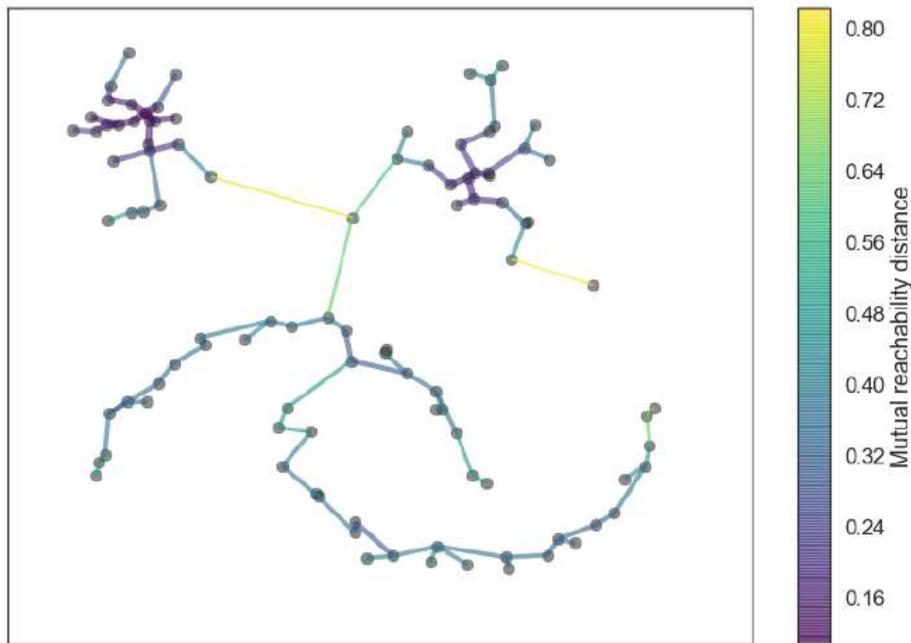


Figura 2.15: Minimum Spanning Tree - MST de todos os pontos conectados por sua distância de alcance mútuo (MCINNES; J; S., 2016)

Os pesos das arestas são reduzidos continuamente, resultando em um grafo desconexo, interligando os diferentes objetos. Um dendrograma pode ser criado para representar a árvore hierárquica com o nó raiz representando todos os pontos em um *cluster* e os níveis subsequentes são divididos em subgrupos até que cada ponto seja um *cluster* em si. A Figura 2.16 ilustra a estrutura hierárquica resultante.

O processamento inicial tem como resultado uma árvore com muitos pontos únicos chamados de (*clusterses singletons*) formando pequenos aglomerados. Para lidar com esses pequenos grupos e gerar mais coesão, o HDBSCAN utiliza como estratégia a redução da árvore inicial, resultando em uma árvore mais simples e compacta (árvore condensada). Essa estratégia é alcançada pela manipulação do parâmetro *min\_cluster\_size*. Similar ao *MinPts*, este parâmetro é utilizado para descartar *clusterses* cujos valores sejam inferiores a um certo limite.

Analogamente, as divisões de um *cluster* na árvore reduzida, acontecem apenas se ambos os eventuais *clusters* filhos contiverem o conjunto que satisfaça o parâmetro *min\_cluster\_size*. Assim, se apenas o *cluster* filho tem objetos suficientes, o *cluster* pai é mantido e o *cluster* filho é incorporado ao *cluster* pai, por outro lado, se nenhum deles tiver objetos que atendam o parâmetro de *min\_cluster\_size*, o *cluster* desaparece. A Figura 2.16 ilustra essa representação.

HDBSCAN é um dos principais algoritmos de agrupamento hierárquico baseado em densidade, disponível na literatura. Em (CAMPELLO et al., 2015), essa abordagem foi estendida para outros domínios, permitindo a geração de agrupamentos por mecanismos não supervisionados, semi-supervisionados, detecção de valores discrepantes e visualização de dados.

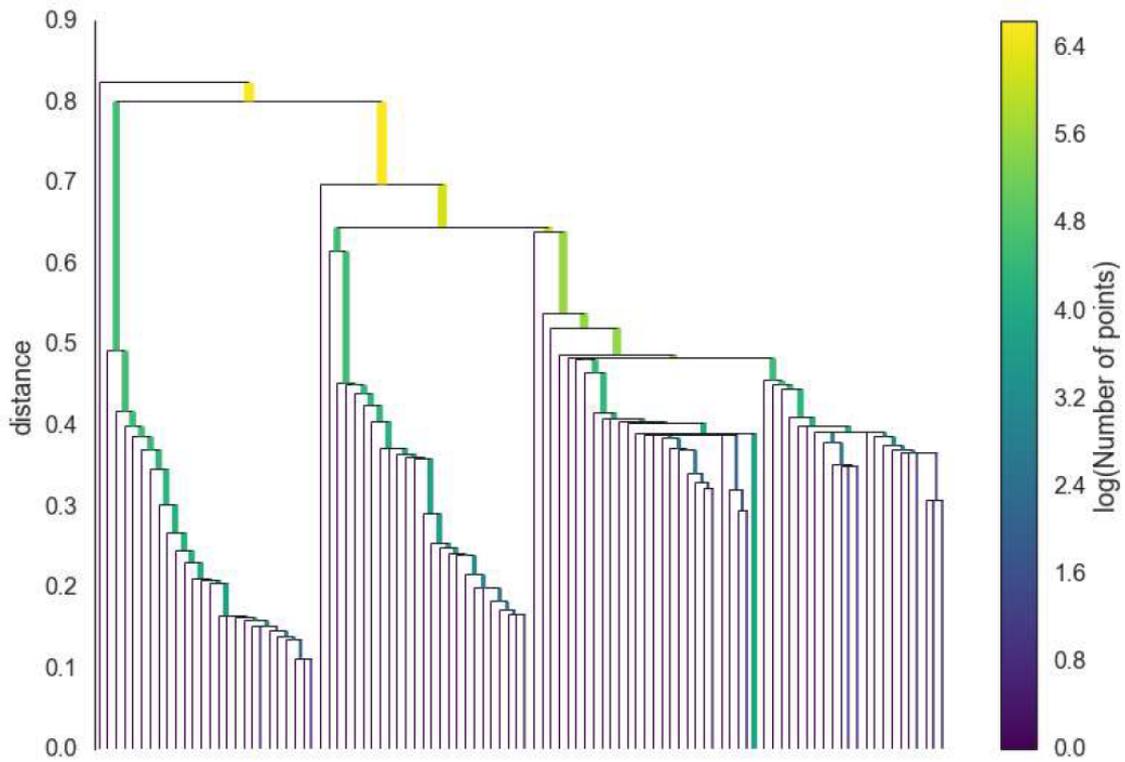


Figura 2.16: Árvore HDBSCAN (MCINNES; J; S., 2016)

#### 2.7.4 X-Means

Embora o algoritmo K-Means seja amplamente usado para detectar agrupamentos, sua eficiência está condicionada a definição prévia da quantidade de grupos ( $k$ -número) a serem retornados. Esta condição impele ao utilizador um certo conhecimento acerca do domínio de aplicação, algo que pode ser desafiador em certos cenários. Uma abordagem para mitigar essa desvantagem pode ser a utilização do algoritmo X-Means.

X-Means foi desenvolvido por (PELLEG; MOORE, 2000), sua abordagem preconiza a localização de um  $K$  arbitrário, conhecido a partir de sucessivas interações. O processo tem início com um  $K = 2$  ou 2-Means (K-Means com  $k = 2$ ) como uma sub-rotina que tenta subdividir repetidamente, mantendo as melhores divisões resultantes. Um ponto de parada é estabelecido quando um Critério de Informação Bayesiano ou *Bayesian Information Criterion* (BIC) (SCHWARZ, 1978) seja alcançado. O BIC é baseado no  $\log$  da probabilidade do conjunto de dados e seu funcionamento é ilustrado na Equação 2.3.

$$BIC(j) = \log M_j(D) - \frac{1}{2}kp_j \log n \quad (2.3)$$

Onde  $M_j(D)$  consiste na probabilidade máxima de  $D$  usando o modelo  $j$  e  $kp_j$  é o número de parâmetros do modelo e  $n = |D|$ . O critério BIC é processado em nível local e global com o fim de identificar não só o melhor número de agrupamentos, mas também, selecionar os grupos de candidatos a serem subdivididos, assim, pode-se definir

essa atuação da seguinte forma:

**Definição 11.** *Seja o conjunto  $D = \{p_1, p_2, \dots, p_n\}$  contendo  $n$  pontos no espaço dimensional  $D$  e seja uma família de modelos alternativos  $M_j = \{M_{k_{Min}}, M_{k_{Min}+1}, \dots, M_{k_{Min}+n}\}$ , com diferentes modelos correspondendo a solução com diferentes valores de  $K$ . Dessa forma, utiliza-se a probabilidade posterior  $P(M_j, D)$  para ranquear os diferentes modelos.*

Durante a execução, as sucessivas interações variando os valores de  $K = K_{Min}$  realizam um incremento do número de centroides nas regiões nas quais um  $K$  adequado não foi identificado, este processo continua até que um  $K$  seja identificado ou que o número máximo de interações seja alcançado. Em ambos os processos, os melhores valores de **BIC** são identificados e ranqueados. Diante disso, pode-se observar que todo o processo de interação pode ser resumido em duas etapas fundamentais, a saber:

1. **Melhorar os parâmetros**, que, em última análise, consiste na execução do K-Means convencional;
2. **Melhorar a estrutura**, que consiste na definição de novos centroides que promovam a subdivisão de agrupamentos já identificados a partir de cada novo  $K$  selecionado.
3. **Se  $k > K_{Max}$** , uma condição de parada foi alcançada, retorna o modelo com melhor valor, caso contrário, retorna ao *Passo 1*.

Os algoritmos de agrupamento analisados neste trabalho foram adaptados para atuar em um contexto de fluxo de dados contínuos. Assim, conforme os dados são recebidos dos ambientes de mobilidade, os mesmos são processados com o fim de identificar o estado atual do tráfego nas diferentes rodovias. Com base nesse panorama do tráfego entre as rodovias monitoradas, torna-se possível identificar os melhores caminhos para se chegar a um certo destino. Para apoiar essa análise, a presente pesquisa se apoia nos algoritmos *k-Shortest path* como forma de identificar os  $K$  melhores caminhos. Essa abordagem é descrita de forma detalhada na próxima seção.

## 2.8 K SHORTEST PATH

Um dos principais desafios de um Sistema de Transporte Inteligente (ITS) é fornecer um mecanismo de re-roteamento capaz de maximizar a eficiência das rodovias por meio de uma distribuição coordenada do fluxo de tráfego. Nestes cenários, pode ser necessário identificar rotas alternativas entre dois pontos, classificando sua prioridade com base em um certo conjunto de pesos e critérios relacionados ao contexto em que cada veículo se encontra, este cenário denota o problema dos  $K$  menores caminhos ou *k shortest path* (KSP).

O problema dos  $k$  menores caminhos consiste em uma generalização do problema do menor caminho, neste caso, ao invés de identificar apenas o menor caminho entre dois

vértices, procura-se ir além, identificando um certo conjunto de possíveis  $k - 1$  caminhos mais curtos, não importando se as demais rotas são mais longas do que um certo  $K$  identificado como menor caminho.

O problema pode ser interpretado da seguinte forma, seja  $G = (V, E)$  um grafo com pesos não-negativos nos arcos,  $s, t$  dois vértices desse grafo e  $k$  um inteiro maior ou igual a 1. Seja  $PG(s, t)$  o conjunto de todos os caminhos de  $s \dots t$  em  $G$ . Assim, o problema dos  $k$  menores caminhos requer a identificação de um subconjunto  $R \in PG(s, t)$  contendo exatamente  $k$  menores caminhos de  $s$  a  $t$ . Ou seja, um conjunto  $R$  tal que não exista um caminho  $P \in PG(s, t)$ , tal que  $P \in R$  e cujo custo seja estritamente menor que algum caminho em  $R$  (KYKUTA, 2018).

Muitos autores se debruçaram sobre este problema, resultando em diferentes propostas disponíveis na literatura, como (BOCK et al., 1957; HOFFMAN; PAVLEY, 1959; DREYFUS, 1969), para citar alguns. Este trabalho utiliza a abordagem proposta por (YEN, 1971) para identificar os  $k$  menores caminhos em um sistema de transporte inteligente. A abordagem proposta por (YEN, 1971) consiste em um método iterativo com o fim de alcançar uma solução para o problema dos  $k$  menores caminhos, sua abordagem preconiza a descoberta de novos caminhos a cada interação. Em termos práticos, o algoritmo deve receber um grafo  $G$  com peso nos arcos vértices  $s$  e  $t$  e um inteiro  $K > 0$  que deve denotar os  $k$  caminhos perseguidos.

No contexto do *framework* DaRkaM, a abordagem proposta por (YEN, 1971) foi adaptada para atuar em uma arquitetura de fluxo de dados contínuos e validada por meio de um conjunto de experimentos e replicações. Nesse contexto, a malha viária foi interpretada com um grafo direcionado, no qual os pesos foram atribuídos conforme a variação da densidade veicular nas rodovias ou arestas. Esta estratégia permite o recálculo dos  $k$  menores caminhos em tempo real, com base no contexto de cada veículo e a situação dos pesos das rodovias a serem utilizadas para a oferta de rotas com foco na melhor eficiência do tráfego. A abordagem proposta por (YEN, 1971) manteve-se por décadas como a mais eficiente e teve suas medidas de desempenho ou performance superadas por (GOTTHILF; LEWENSTEIN, 2009) apenas no ano de 2009.

## 2.9 CONSIDERAÇÕES FINAIS

Este capítulo abordou os princípios fundamentais que nortearam o desenvolvimento da presente pesquisa. Entre os temas abordados nesta seção estão os fundamentos do emergente paradigma da computação em névoa ou *Fog Computing*. Nesta seção, foi descrito o papel da névoa, seus elementos e como a complementaridade entre névoa e nuvem pode trazer benefícios neste novo universo de dispositivos conectados em franca expansão. De maneira complementar, abordou-se nesta seção ainda, o papel da redução de dados nos sistemas de transporte inteligente, suas motivações e técnicas que, de certa forma, nortearam ou contribuíram para a realização desta pesquisa.

Ao longo desta seção, foi possível identificar como a implementação de estratégias de redução na névoa podem reduzir significativamente o tráfego de dados para a nuvem, levando de forma concomitante à redução nos custos de comunicação. Em seguida, foram abordadas as redes veiculares e sua relação com o paradigma da névoa. As redes veicu-

lares cumprem um papel essencial na construção de Sistemas de Transporte Inteligente e quando associadas ao paradigma da computação em névoa, novos e mais eficientes sistemas podem ser modelados. Adicionalmente, abordou-se neste capítulo ainda, o papel dos algoritmos de agrupamento, em especial agrupamentos baseados em densidade espacial.

As técnicas de redução de dados propostas neste trabalho são, em essência, baseadas no agrupamento espacial dos elementos, analisando como essas diferentes abordagens podem ser combinadas para produzir resultados positivos em aspectos como a redução de dados e dos custos de comunicação em um sistema de transporte inteligente. Na perspectiva da acurácia dessas abordagens, este capítulo abordou o manual de capacidade de tráfego e o conceito de classificação baseada no nível de serviços. A identificação do nível de serviço tem grande relevância não só na identificação de congestionamentos, mas também na identificação previa dos mesmos, a partir do monitoramento das variações de classe em um certo cenário urbano. Por fim, foram abordadas estratégias de roteamento apresentando a abordagem *k shortest path* e como ela pode ser aplicada ao contexto dos sistemas de transporte inteligente.

Espera-se que estas seções tenham oferecido uma visão consistente acerca dos elementos, técnicas e tecnologias utilizadas na confecção desta pesquisa. O próximo capítulo irá abordar o estado da arte da computação urbana por meio de uma revisão dos trabalhos correlatos.

## **TRABALHOS RELACIONADOS**

A computação urbana persiste como uma subárea das cidades inteligentes e preocupa-se com a orquestração de seus elementos para superar alguns desafios da mobilidade urbana. Nesse contexto, um desafio emerge com grande relevância, a construção de eficientes sistemas de gestão e monitoramento do tráfego capazes de atuar no cenário urbano maximizando o desempenho do fluxo nas rodovias monitoradas. As próximas seções apresentam um panorama acerca dos estudos que abordam os elementos que compõem um sistema de transporte inteligente, combinando diferentes visões e estratégias com ênfase nas questões de tráfego e mobilidade urbana.

A análise dos trabalhos correlatos está subdividida da seguinte forma: A Seção 3.1 fornece uma visão geral acerca do processo de detecção de congestionamento; em seguida, a Seção 3.2 ilustra o uso das mídias sociais como um meio para alavancar melhorias de mobilidade através da obtenção de dados de posicionamento global. A Seção 3.3 apresenta o uso de sistemas preditivos em redes veiculares híbridas. Na Seção 3.4, são apresentados estudos que utilizam estratégias baseadas em similaridade e agrupamento para determinar a situação do tráfego; Na Seção 3.5, são apresentados estudos que abordam a redução de dados e seus impactos nos custos de comunicação; Subsequentemente, na Seção 3.6, são analisadas as correlações entre os trabalhos analisados e, por fim, na Seção 3.7, são realizadas algumas considerações acerca desse capítulo.

### **3.1 DETECÇÃO DE CONGESTIONAMENTO EM SISTEMAS DE TRANSPORTE INTELIGENTE**

Os desafios na identificação do congestionamento de tráfego são potencializados em função das suas características não lineares e sua natureza não determinística (HENRY et al., 2001; YOUNAS; RAUF; SYED, 2006; CÁRDENAS-BENÍTEZ et al., 2016; SOUZA et al., 2016; SOUZA; VILLAS, 2016; RANI; SHAW, 2019; AKABANE et al., 2020). Diante disso, diferentes abordagens têm sido empregadas por pesquisadores ao redor do globo com o fim de oferecer contribuições, sejam relacionadas à detecção de congestionamento propriamente dito, sejam nos aspectos correlacionados, como a redução de dados e redução dos custos de comunicação.

Um importante passo para realizar a detecção de congestionamento está relacionado ao conhecimento acerca das posições, velocidade média e comportamento dos veículos em um certo raio de cobertura. Nesse contexto, diferentes abordagens têm sido empregadas, a exemplo do monitoramento de dados a partir de unidades estacionárias como *RoadSide Unit* ou *Cloudlets*, ou ainda, abordagens mais recentes como *Floating Car Data* (FCD) que, ao contrário dos métodos tradicionais que analisam o fluxo contínuo de dados a partir de um dispositivo fixado na rodovia, adotam uma abordagem mais distribuída, utilizando-se essencialmente da comunicação V2V (veículo para veículo) para realizar este monitoramento.

Nesse cenário, uma abordagem V2V foi proposta por Mohanty et al. (2022). Em sua proposta, os autores descrevem um cenário no qual os veículos são equipados com um sistema de comunicação sem fio que garante a comunicação entre veículos vizinhos ao longo do tráfego. Entretanto, os autores argumentam que a troca de informação em um sistema V2V, no qual veículos vizinhos podem trocar informações livremente, em certos cenários pode impor significativo impacto na largura de banda dessas redes e, para mitigar esses desafios, se utilizam de um sistema de redução de dados baseados em agrupamento. Nessa abordagem, a detecção de congestionamento se dá por meio da média dos dados coletados entre os veículos, os quais são agrupados por faixas das rodovias analisadas e submetidos a análise por meio de um *Analytical Hierarchy Process* (AHP), responsável por fornecer dados que denotem os níveis de congestionamento das faixas.

Analogamente, em Wahid, Rao e Goel (2019) os autores utilizam dados de Fluxo Contínuo de Dados (FCD) e *Global Positioning System* (GPS) para reduzir o atraso de comunicação entre os veículos e o servidor de uma rede. Para fazer isso, os autores propõem um método de detecção de congestionamento considerando que todos os veículos operam como nós sensores, enviando dados para um servidor. Esse servidor, por sua vez, recebe informações de tráfego para calcular o congestionamento da via. Após o processamento, um retorno é enviado a todos os usuários da rodovia. Para definir quais dados seriam enviados, os autores propõem uma política de transmissão que não usa critérios de randomização ou custo de informação. Nessa abordagem, todos os veículos da rede têm a mesma oportunidade de transmitir dados para o servidor.

Utilizando uma estratégia de comunicação híbrida, Ramachandra et al. (2016) propõem a utilização de sensores *On Board Unit* *On-Board Diagnostic* (OBD) com o fim de medir a densidade do fluxo de tráfego em uma determinada rodovia. Nesta abordagem, os veículos são equipados com um dispositivo OBD que tem a função de obter os dados de velocidade da infraestrutura interna veicular e transmiti-los por meio de uma conexão *bluetooth* para uma unidade central denominada *On Board Unit* (OBU). Uma vez no repositório, os dados são enviados para a infraestrutura da rodovia.

Com uma abordagem mais robusta, Rath, Pati e Pattanayak (2019) propõe um agente móvel embutido em um microcontrolador para detectar congestionamento de tráfego e sugerir intervenções no cenário urbano com o fim de aumentar a eficiência no fluxo de tráfego. O projeto estrutura-se em um conjunto de três módulos que atuam por meio do monitoramento de vídeo, controle de tráfego e sistemas de comunicação. A abordagem proposta pelos autores demonstrou redução no tempo de detecção de congestionamento quando comparada a outras abordagens como o sistema de gerenciamento de tráfego

dinâmico *Dynamic Traffic Management System* (DTMS) ou sistemas de detecção tradicionais.

Diferentemente das abordagens apresentadas até aqui, para modelar ou estimar as condições de tráfego, em (KUMAR; KUSHWAHA, 2016; KUMAR; KUSHWAHA., 2019), os autores propõem uma estratégia de detecção de congestionamento de tráfego com base em imagens capturadas de câmeras instaladas em diferentes locais de uma certa cidade. A abordagem concentra-se em dois fatores de forma concomitante, a identificação da quantidade de veículos observados na imagem e a largura da rodovia. Com base nestes dados, um algoritmo é capaz de estimar a densidade do tráfego na rodovia.

De maneira similar, autores em (TSUYOSHI et al., 2017) propõem em seu trabalho a detecção de congestionamento de tráfego por meio da análise de eventuais mudanças ocorridas nas imagens ao longo do tempo. A proposta dos autores versa sobre o emprego de um algoritmo de aprendizado de máquina não supervisionado capaz de quantificar o número de veículos em uma imagem, e assim, aferir os níveis de congestionamento. A detecção de congestionamento baseado na análise de imagens enfrenta alguns desafios que vão desde a baixa capacidade dos dispositivos disponíveis nas rodovias até eventos de ordem natural como as condições climáticas ou poluição.

Trabalhos como (PALUBINSKAS; KURZ; REINARTZ, 2008; CHENG; LIU; TANG, 2016; LAM; GAO; NG, 2017; RAO et al., 2018; LI; WANG; BUCKLES, 2019; SATYA-NARAYANA; MAJHI; DAS, 2021) apenas para citar alguns, alertam para os desafios na identificação de congestionamento de tráfego baseado em imagem sequenciais. Autores argumentam que os aspectos ambientais como condições de chuva ou neblina podem reduzir substancialmente a utilização desta abordagem. Adicionalmente, a abordagem perde escalabilidade na medida em que o tratamento das imagens é realizado em uma infraestrutura local. Além disso, a transferência deste processamento para a nuvem afetaria significativamente os custos de comunicação, além de reduzir a confiabilidade da latência.

Os estudos apresentados, incluindo (XU; YUE; LI, 2013; BAUZA; GOZÁLVEZ, 2013; ANJUM; BADRUDDIN; DRIEBERG, 2014; LV et al., 2014; WANG; WORK; SOWERS, 2016; ABRAHAM; WELLER, 2017; D'ANDREA; MARCELLONI, 2017; AHMADZADEGAN et al., 2019; DIMRI et al., 2019), não analisam os efeitos da aquisição dos dados em termos de custos de infraestrutura, distribuição de sensores em segmentos de uma rodovia ou os efeitos provenientes na análise massiva de dados pela nuvem e seus impactos nos custos de comunicação. Além disso, os trabalhos analisados são alheios ao custo de comunicação associado à troca de dados entre veículos e infraestrutura durante os processos de monitoramento e detecção de congestionamento.

### **3.2 SISTEMAS DE TRANSPORTE INTELIGENTE BASEADOS EM MÍDIAS SOCIAIS**

As mídias sociais também têm sido observadas como um meio promissor no suporte ao gerenciamento de tráfego e sistemas de transporte inteligente. As grandes quantidades de dados trafegados por essas ferramentas compõem uma ampla rede de conhecimento, capaz de contribuir para mitigar alguns desafios da computação urbana, entre eles o

monitoramento, identificação de congestionamento ou ainda a sugestão de novas rotas em função do tráfego (RODRIGUEZ; FERREIRA; POCO, 2022).

A velocidade de transmissão das informações em mídias sociais é certamente o ponto-chave para que esta seja considerada promissora. Em linhas gerais, estas redes respondem, em poucas palavras, questões complexas como: “*O que está acontecendo, onde e quando?*”. Responder estas perguntas pode ser um passo importante para identificar o padrão de mobilidade dos usuários e por consequência, ser mais efetivo na análise e predição do comportamento do tráfego, fornecendo maiores subsídios para a tomada de decisão. No entanto, assim como nas abordagens apresentadas, essa também não é uma tarefa trivial.

Muitas ferramentas nas redes sociais utilizam marcadores de localização, a exemplo do *Twitter*, *Foursquare* ou *Facebook*. Nestas ferramentas, as postagens são, em geral, associadas a uma determinada localização. Baseado nessa observação, (WANG et al., 2019b) propõem um método de detecção de atrasos ferroviários que utiliza o aplicativo *Twitter*. Em sua abordagem, os autores propõem uma análise considerando uma taxa de propagação dos tópicos de *tweets* com *geo-tag* entre as estações ferroviárias (WANG et al., 2019b).

O objetivo deste modelo é descobrir ou prever atrasos de trens nas estações subsequentes. A eficiência desta abordagem centra-se em dois aspectos fundamentais *a)* não requer sensores ou infraestrutura de detecção adicional; *b)* a inexistência de alterações nos custos de comunicação. Por outro lado, o uso de redes sociais exige acesso a grandes volumes de dados públicos, além de alta capacidade de processamento e mineração de dados em texto. Esses aspectos não são avaliados no trabalho.

Em outro trabalho, Wang et al. (2019a) utiliza informações de eventos de tráfego extraídos das mídias sociais com o fim de combiná-los com dados de localização espacial. Para os autores, o cruzamento destas informações permite a realização de estimativas no que se refere a identificação de congestionamento de tráfego. A abordagem proposta pelos autores estende o modelo Markov (*Coupled Hidden Markov Model*) com o objetivo de lidar com as questões de heterogeneidade entre esses tipos de dados. Os autores não abordam os custos de comunicação advindos da integração entre sistemas de posicionamento e os dados de mídia sociais. Do mesmo modo, não são mencionados aspectos arquiteturais dessa solução.

Adicionalmente, em Rodriguez, Ferreira e Poco (2022) os autores propõem uma abordagem denominada *JamVis*, com o objetivo de estabelecer um modelo capaz de maximizar o entendimento do comportamento do tráfego a partir de um sistema de visualização de dados de tráfego estruturado sob a aplicação *Waze*. Para os autores, aplicações baseadas em GPS se popularizaram ao longo dos últimos anos e revolucionaram a forma como os humanos se relacionam com o trânsito e a mobilidade. Neste contexto, a aplicação *Waze* cumpre um relevante papel, para os autores, aplicações como *Waze* fornecem uma enorme quantidade de dados que podem ser analisados sob diferentes perspectivas, seja na análise do tráfego em uma perspectiva espaço-temporal, seja na captura de alertas emitidos pelas contribuições dos seus milhares de usuários. Baseado nessa perspectiva, os autores definem em seu trabalho um sistema de visualização que contempla duas principais contribuições *a)* um sistema de identificação de eventos e alertas baseado em agrupamento, denominado *ST-DBSCAN* e *b)* um sistema de visualização baseado em mapa de calor

denominado *ST-Heatmap*, capaz de denotar o comportamento do tráfego ao longo do tempo.

O uso de redes sociais como fonte de dados para os sistemas de transporte inteligente surge como uma estratégia promissora para a detecção de congestionamento de tráfego, em especial, porque tais abordagens não requerem o uso de uma infraestrutura de monitoramento, além de atuarem em um sistema de colaboração onde o engajamento torna-se o fator prioritário para alcançar melhores níveis de acurácia. Entre os trabalhos analisados não constam detalhes acerca dos custos de processamento, latência ou redução de dados, foco principal da presente pesquisa.

### 3.3 PREDIÇÕES NO TRÁFEGO UTILIZADO SISTEMAS HÍBRIDOS

Redes veiculares ou VANET's (*Vehicular Ad hoc NETWORKS*) oferecem certa versatilidade no que se refere a sua topologia e como os nós de uma certa rede interagem entre si. Esses novos modelos de comunicação aliados aos novos padrões arquiteturais viabilizados por paradigmas como Névoa e Nuvem abrem espaço para a construção de sistemas que, além do monitoramento, atuam na prevenção do comportamento do fluxo de tráfego. O conhecimento prévio acerca deste comportamento pode contribuir significativamente para decisões como o redirecionamento de tráfego ou sugestões de novas rotas aos condutores.

Diante desse desafio, Rani e Shaw (2019) propõem uma abordagem híbrida, envolvendo níveis de comunicação veículo para veículo (V2V) e veículo para infraestrutura (V2I). Em sua proposta, os autores se utilizam da lógica *fuzzy* para estimar os níveis de congestionamento de tráfego em uma certa rodovia. Nesse trabalho, os autores empregam atributos técnicos como o volume de tráfego, tempo de luz verde, tempo de ciclo e os combinam com atributos não técnicos como as condições do clima, as condições da estrada, visibilidade, entre outros. Esses conjuntos de dados são então fornecidos como entrada no processo de agregação de regras baseadas em lógica *fuzzy*. Como resultado, tem-se uma estimativa do fluxo de tráfego nas rodovias.

Embora os autores estimem os atrasos do tráfego nas interseções de uma rede rodoviária, o estudo carece de análises mais acuradas acerca dos seus impactos nos custos de comunicação que estão, em certa medida, intrínsecos no processo de aquisição dos dados disponibilizados pela malha de rede móvel veicular. Além disso, os autores não apresentam elementos para análise dos impactos do volume de dados em termos de sobrecarga da infraestrutura ou das próprias interações V2V promovidas pela abordagem *Floating Car Data* (FDC).

A lógica *fuzzy* foi aplicada também por Chen e Li (2013). Nesse trabalho, os autores propõem um estudo da estimativa da fase do fluxo de tráfego com base nos dados obtidos a partir de uma rede veicular VANET. O estudo concentra-se na classificação dos níveis de fluxo de tráfego considerando três estados fundamentais: fluxo livre, fluxo sincronizado e fluxo congestionado. Nesse estudo, o processo de classificação é realizado por meio de um algoritmo de predição baseado em lógica *fuzzy*. Os experimentos foram estruturados sobre uma rede de simuladores que incorpora elementos como o SUMO para representação dos aspectos de mobilidade e NS2 para simulação das redes veiculares VANET. Trabalhos como (CHEN; LI, 2013; BAUZA; GOZALVEZ; SANCHEZ, 2010) e

(BAUZA; GOZÁLVEZ, 2013), também utilizaram a lógica *fuzzy* como forma de apoiar a identificação e/ou predição do congestionamento de tráfego.

Uma abordagem mais recente foi proposta por (ALKHEDER; ALOMAIR, 2021) ao analisar o sistema de transporte do Kuwait estabelecendo uma relação entre os níveis de congestionamento e as mudanças meteorológicas da região. A abordagem se utiliza de algoritmos de aprendizado de máquina e estratégias baseadas na lógica *fuzzy* para estabelecer um sistema de predição. Experimentos iniciais indicaram uma relação clara entre o tráfego e a sua associação com variáveis meteorológicas, das quais a temperatura foi a variável mais representativa. Neste trabalho, três modelos de aprendizado de máquina foram utilizados, dos quais o modelo *long short-term memory* (*Long Short-Term Memory* (LSTM)) manteve desempenho superior aos demais.

Uma outra abordagem preditiva também foi proposta em (ZHANG et al., 2020). Nesse trabalho, os autores apresentam uma abordagem que tenta combinar aspectos da redução no uso dos recursos computacionais com o aumento no desempenho na realização de análises preditivas para identificação de congestionamentos. Para isso, os autores se utilizam de uma rede neural treinada a partir de um conjunto de dados históricos obtidos em um certo cenário urbano. A eficiência deste modelo foi mensurada a partir de um recorte bruto do cenário de tráfego sob o qual um subconjunto de dados é obtido e processado com o fim de identificar as variações na densidade veicular nas rodovias. As medidas de acurácia consideram o uso de uma abordagem proposta pelos autores denominada *Maximum Value in Each Grid* (MAV) para comparação das predições e mensuração dos níveis de acurácia alcançados.

Adicionalmente, os estudos de (BAUZA; GOZÁLVEZ; SANCHEZ, 2010) e (BAUZA; GOZÁLVEZ, 2013) propõem a utilização da técnica CoTEC (*Cooperative Traffic congestion detection*), que visa otimizar a detecção de tráfego de veículos sem a instalação de sensores de infraestrutura, utilizando apenas comunicação V2V. Para detectar as condições de congestionamento, CoTEC usa um mecanismo de lógica *fuzzy* baseado em mensagens de sinal recebidas de veículos vizinhos.

Em (BAUZA; GOZÁLVEZ; SANCHEZ, 2010), os autores conduziram e avaliaram uma simulação de congestionamento utilizando o simulador de tráfego (SUMO) e o CoTEC em ambiente rodoviário. Os parâmetros de velocidade e localização foram analisados. O alcance do comunicador empregado foi de 300 metros, que transmite mensagens periodicamente a 1 Hz, sinalizando a localização e velocidade do veículo. Para garantir maiores níveis de precisão dos dados da intensidade do tráfego é necessária uma certa quantidade de veículos retransmissores. Quanto maior o número de retransmissores, maior a qualidade da informação e menor o *Root Mean Square Error* (RMSE).

Adicionalmente, em (BAUZA; GOZÁLVEZ, 2013) os autores utilizam o iTETRIS, um simulador *opensource*, que foi desenvolvido para investigar o impacto de sistemas veiculares cooperativos. Os resultados demonstraram que a solução apresentou altos níveis de acurácia na representação do estado do tráfego durante as simulações. A abordagem consiste na captura e compartilhamento dos dados de tráfego em tempo real. Os experimentos foram realizados utilizando um *middleware* capaz de integrar um conjunto de tecnologias. No contexto das simulações foram utilizadas as ferramentas SUMO e NS3.

Os trabalhos analisados nessa seção abordam estratégias de predição em um sistema

de comunicação VANET híbrido, no entanto, nenhuma das abordagens foi aplicada em um contexto de fluxo de dados contínuo. Além disso, não existem detalhes acerca da eventual redução de dados decorrente das abordagens descritas. O *framework* proposto neste trabalho procura lidar com esse desafio, a partir de uma estrutura de processamento adaptável capaz de utilizar diferentes estratégias de redução de dados em um contexto de fluxo de dados contínuo.

### 3.4 TÉCNICAS DE AGRUPAMENTO APLICADAS AO CENÁRIO URBANO

Técnicas de agrupamento têm sido usadas para particionar um grande número de dados veiculares heterogêneos em diferentes grupos de dados homogêneos. Com isso em mente, Lai et al. (2007) propõem um método baseado em agrupamento cujo o objetivo é processar análises baseadas em densidade a partir de uma rede rodoviária. A abordagem proposta pelos autores calcula a densidade usando parâmetros como comprimento e número de objetos no segmento de estrada. Nesse estudo, um *cluster* é definido como um grupo de objetos em movimento contínuo, próximos uns dos outros.

Outra abordagem também baseada em agrupamento/*cluster* foi proposta em (AMELIA; SAPTAWATI, 2014). Nesse trabalho, os autores aplicam técnicas de mineração de dados espaço-temporal para detectar potenciais congestionamentos de tráfego nas estradas. No entanto, os autores não analisam o custo de comunicação nem identificam a origem do congestionamento de tráfego. Adicionalmente, os autores não utilizam esses resultados na tomada de decisão para o aprimoramento inteligente do fluxo de tráfego.

Similarmente, em Zhang et al. (ZHANG et al., 2016), os autores propõem um algoritmo de agrupamento para analisar o grau de congestionamento de tráfego baseado na teoria de análise relacional cinza. Esta abordagem se utiliza de dados multidimensionais que são convertidos em diferentes níveis de agrupamento. Os autores utilizam três variáveis de fluxo de tráfego, quais sejam: velocidade, densidade e volume para avaliar o grau de congestionamento do tráfego. O trabalho de (ZHANG et al., 2016) foi fundamental para a elaboração da presente proposta, uma vez que serviu como inspiração para a construção de um modelo arquitetural para monitoramento, detecção e tomada de decisão nos cenários urbanos.

Da mesma forma, o trabalho de (MOHANTY et al., 2022) aplicou o algoritmo de agrupamento K-Means para análise de dados em uma estrutura de comunicação V2V. Nesta abordagem, os autores coletam dados oriundos da troca de mensagens entre os veículos com o fim de identificar a presença de congestionamentos nas faixas das rodovias analisadas. Em sua abordagem, uma média aritmética dos parâmetros contidos nas mensagens fornece indícios da presença de congestionamentos. Estes dados são então classificados por um sistema de priorização denominado *Analytical Hierarchy Process* (AHP). Dados relativos ao consumo de rede ou detalhes acerca do sistema de comunicação não são fornecidos neste trabalho.

Adicionalmente, um esquema de agregação de tráfego em tempo real, usando a propriedade de recuperação de dados na assinatura de recuperação de mensagem *Message Recovery Services* (MRS) foi proposto em (SHEN et al., 2019). Esse estudo enfatiza questões relacionadas aos recursos de segurança dos dados, como resistência a ataques,

preservação da privacidade e sigilo dos dados. Nesse trabalho, os autores consideraram o uso de autenticação na transmissão de mensagens, realizando a validação e verificação das assinaturas dos veículos na rede, de forma a garantir a segurança e evitar que os dados sejam expostos a possíveis ataques. Adicionalmente, os autores propuseram ainda a aplicação de operações em lote, aumentando a eficiência na verificação de assinaturas de múltiplos veículos.

Em uma abordagem similar, os autores em (GUEDES; CAMPOS, 2016) apresentam um esquema de agregação de dados que visa reduzir a quantidade de dados redundantes em uma VANET com o objetivo de mitigar problemas relacionados à escalabilidade. Nesse estudo, (GUEDES; CAMPOS, 2016) propõe a utilização de dados agregados, aplicando uma estratégia baseada em caminhos de comprimento fixo, que por sua vez, são usados como parâmetro para o mecanismo de decisão. Para validar a aplicação do esquema, o simulador de redes veiculares NCTUns foi usado neste estudo.

Outro modelo de agregação foi proposto em (KAUR; KAD, 2016). Neste, os autores propõem um modelo de agregação de dados que utiliza o protocolo *Adhoc on Demand Distance Vector Reliability* (AODV-R) baseado na otimização de colônias de formigas. O protocolo utilizado é reativo, assim, um link entre os nós de envio é estabelecido apenas quando necessário. Para melhorar o processo de escolha do caminho mais curto, os autores propuseram a utilização do algoritmo de otimização *Ant Colony*, substituindo os algoritmos *Dijkstra* e *hop by hop* implementados no protocolo AODV-R. O trabalho não considera a falha do nó e não descreve os elementos arquiteturais sob os quais a proposta foi concebida, seja em *Fog* ou *Cloud*.

Um algoritmo de agrupamento baseado em otimização de colônias de formigas para redes de veículos denominado CACONET foi proposto também em (AADIL et al., 2016). Essa proposta tem como objetivo otimizar a formação de *clusters* para obter uma comunicação robusta entre os componentes de uma VANET. A ideia centra-se em minimizar o número de *clusterhead* na rede, a fim de obter uma redução no custo de comunicação. Para validar o uso do algoritmo, os autores realizaram uma comparação com outras técnicas de otimização baseadas na redução do número de *clusterhead* como: *Multi-Object Particle Swarm Optimization* (MOPSO) e *Comprehensive Learning Particle Swarm Optimization* (CLPSO).

Adicionalmente, em (SHAH et al., 2022) os autores propõem um mecanismo baseado em agrupamento como forma de reduzir os índices de acidentes nas rodovias por meio do monitoramento do tráfego. Em sua abordagem, os autores apresentam um algoritmo baseado *Moth-Flame Optimization* (*Moth-Flame Optimization* (MFO)), denominado AMONET. Os experimentos foram avaliados frente a outras propostas da literatura como CLPSO, MOPSO e CACONET apresentando resultados promissores que tangenciam benefícios como a redução do número de agrupamentos, redução de dados e maximização do desempenho geral da rede.

Conforme mencionado, algoritmos de agrupamento são eficientes para analisar padrões, formar grupos de dados, auxiliar na tomada de decisões e aprendizado de máquina, incluindo mineração de dados e classificação de padrões. Diante disso, em (SILVA et al., 2014) os autores propõem o uso do algoritmo DBSCAN em uma abordagem distribuída usando o *MapReduce* com o fim de identificar áreas de congestionamento em dados de

mobilidade urbana. Os experimentos apresentados demonstram que a estratégia proposta encontrou os mesmos grupos gerados pelo DBSCAN centralizado, porém gastou menos tempo para processar os dados.

Analogamente, a abordagem utilizada por (DING et al., 2015) considera que cada grupo criado pelo DBSCAN pode ter  $\epsilon$  e  $minPts$  diferentes, ou seja, múltiplos parâmetros. Esta abordagem teve por objetivo detectar anomalias no tráfego de rede e os resultados mostraram que a taxa de detecção é maior e a taxa de falsos positivos é menor do que outros algoritmos que assumem as mesmas suposições sobre o conjunto de dados. Essas pesquisas não avaliam como os parâmetros dos algoritmos de agrupamento interferem na composição dos *clusters*.

Uma análise acerca do desempenho do algoritmo DBSCAN na detecção de congestionamento foi também apresentada em (PEIXOTO et al., 2020). Neste trabalho, os autores propõem uma análise exploratória com o fim de identificar os impactos gerados pela variação no raio de vizinhança ( $\epsilon$ ) nas medidas de acurácia em um sistema de detecção de tráfego. Os resultados da avaliação demonstram que há uma relação de dependência entre o tamanho do raio configurado no algoritmo DBSCAN, a quantidade de dados reduzidos e o nível de acurácia na detecção de congestionamentos em um certo cenário urbano.

Conforme é possível observar em trabalhos anteriores, muitos estudos utilizam técnicas de agrupamento como forma de identificar similaridades que conduzam à identificação dos pontos de congestionamento de tráfego e a consequente tomada de decisão, entretanto, poucos lidam com questões relacionadas ao fluxo de dados contínuo. Em geral, os experimentos relatados consideram um conjunto de dados estático e, portanto, não representam cenários viáveis no mundo real. Além disso, os trabalhos, em geral, não consideram os impactos das suas estratégias nos custos de comunicação.

### 3.5 A REDUÇÃO DE DADOS E OS CUSTOS DE COMUNICAÇÃO

Em redes veiculares, um desafio comum tem sido lidar com a enorme quantidade de dados trafegados. Veículos, infraestrutura e demais dispositivos dispostos no cenário urbano competem pelos escassos recursos disponíveis em uma realidade que tende a se acentuar nos próximos anos. Diante disso, as estratégias de redução de dados têm o fim de mitigar o aumento progressivo nos custos de comunicação que ocorrem quando o aumento do uso da infraestrutura tem impacto direto e proporcional na elevação destes custos. Assume-se aqui que as estratégias de redução de dados propostas neste trabalho são um meio de reduzir esta relação de proporcionalidade.

Com o objetivo de minimizar o uso da rede durante os processos de monitoramento e gerenciamento do tráfego, em (Kerner et al., 2005; TANIZAKI; WOLFSON, 2007; WAHID; RAO; GOEL, 2019) os autores propõem uma política de comunicação voltada à redução do número de transmissões dos veículos para o servidor. Nesta abordagem, um veículo poderia decidir o momento do envio da mensagem, por exemplo, em relação ao tempo de viagem em um dado segmento de estrada para um servidor. Esse repositório central então coleta todos os dados de diferentes veículos e usa essas informações nos processos de detecção e gerenciamento de congestionamento de tráfego.

Alternativamente, para enfrentar o problema de redução de dados é possível considerar

uma trajetória espacial como um movimento de um veículo em um espaço geográfico. Geralmente, uma trajetória espacial é representada por uma série de pontos ordenados cronologicamente, como, por exemplo:  $p_1 \rightarrow p_2 \rightarrow \dots \rightarrow p_n$ , em que cada ponto mantém um registro de data e hora e um par de coordenadas geográficas  $(x, y)$ . Uma trajetória é considerada como um fluxo de dados, pois uma série de pontos em movimento contínuo configura uma grande amostra de pontos.

Nesse contexto, os trabalhos de (CERQUEIRA et al., 2018), (SOUZA et al., 2018), (CHATRAPATHI et al., 2015), (IRREVALDY; SAPTAWATI, 2017), (MANJORO; DHAKAR; CHAURASIA, 2016) utilizaram a trajetória espacial de veículos para identificar congestionamentos de tráfego ao longo do tempo em várias estradas, realizando a agregação de dados por meio da previsão do deslocamento veicular nas vias. Entretanto, esses estudos não consideram o impacto na construção dos *clusters* por meio de uma análise exploratória dos parâmetros e métodos de agregação.

Com uma abordagem diferente, os autores em (TANIZAKI; WOLFSON, 2007) propõem uma estratégia de redução dos custos de comunicação por meio de uma função de atualização aleatória. Nessa função, os veículos transmitem sua velocidade ao servidor com uma certa probabilidade  $p$ . Seleções aleatórias também foram utilizadas em (HE; ZHANG, 2017). Nesse estudo, a estratégia consiste em escolher, transportar ou encaminhar pacotes de dados com base nas informações de tráfego em um dado instante de tempo. Nesse trabalho, os autores formularam a coleta de dados como um problema de otimização de escalonamento, usando um algoritmo de programação dinâmica ideal para coleta de dados em pequena escala, além de heurísticas baseadas em algoritmos genéticos para coleta de dados em larga escala.

Outra abordagem foi utilizada em (OLIVEIRA; KNISS, 2019), neste trabalho, os autores apresentam um algoritmo para análise de dispersão dos dados coletados por uma rede de sensores com o objetivo de identificar os valores com baixa variação. Dessa forma, o volume de dados gerados nesse ambiente é reduzido devido ao corte dos dados com alguma similaridade.

Em um viés contrário a aleatoriedade apresentada por (OLIVEIRA; KNISS, 2019), em (Kerner et al., 2005), os autores apresentam uma abordagem determinística, na qual veículos enviam seus dados de velocidade para um servidor apenas se seus valores excederem um limite preestabelecido. Esse limite é conhecido pelos veículos que utilizam a rede veicular.

Analogamente, em (WAHID; RAO; GOEL, 2019) propõe-se uma política de atualização de servidor, onde um veículo envia seus dados para o servidor apenas se a diferença entre a velocidade obtida e a velocidade já armazenada for maior ou igual a certo limite durante algum intervalo de tempo ( $\Delta t$ ).

Em virtude da adaptabilidade desta abordagem para o contexto do framework DaRkaM, proposto neste estudo, os algoritmos propostos por (Kerner et al., 2005; TANIZAKI; WOLFSON, 2007; WAHID; RAO; GOEL, 2019) foram implementados como forma de estabelecer uma base comparativa entre as técnicas de monitoramento e detecção de congestionamento de tráfego presentes entre as contribuições propostas neste trabalho.

Adicionalmente, uma abordagem para redução de dados na névoa foi proposta por (PEIXOTO et al., 2021), onde os autores realizam uma análise comparativa utilizando duas

estratégias de agrupamento em um contexto de fluxo de dados contínuo. Para ilustrar a abordagem proposta, os autores propõem adaptações nos algoritmos *Ordering Points to Identify the Clustering Structure* (OPTICS) e *Density-Based Spatial Clustering of Applications with Noise* (DBSCAN) como forma de integrar a sua atuação ao contexto de fluxo de dados contínuos, permitindo que seu processamento ocorra de forma sistemática com base em pequenos e contínuos conjuntos de dados recebidos.

Os resultados obtidos no referido estudo demonstraram significativa redução nos dados mantendo altos níveis de representatividade, mesmo em cenários altamente congestionados. A abordagem indica uma potencial estratégia para redução de dados e consequente redução dos custos de comunicação e armazenamento. Ambas as estratégias também são utilizadas neste trabalho.

Cabe citar ainda que o referido trabalho foi incluído em uma ampla análise realizada por (SADRI et al., 2022), onde os mesmos analisam estratégias de redução de dados, abarcando em seus estudos os diferentes cenários e pontos de aplicação dessas abordagens. Nesse trabalho, Sadri et al. (2022) realizam uma análise comparativa ilustrando não só os aspectos relativos à comunicação e conectividade, mas também, aspectos como relacionados a acurácia, arquitetura, aplicação e custos. Desde essa publicação, a presente tese traz avanços no processo de redução de dados na medida em que novos algoritmos e técnicas foram implementadas no framework, permitindo a realização de análises comparativas enfatizando aspectos como a detecção de tráfego, análise da acurácia em fluxo contínuo, intervenções de tráfego, além das análises relacionadas ao uso dos recursos de rede e custos de comunicação.

### 3.6 ANÁLISE DA PROPOSTA EM RELAÇÃO À LITERATURA RELACIONADA

Os trabalhos apresentados nesse capítulo foram utilizados como base fundamental para a presente proposta de pesquisa, além de contribuírem para definir o posicionamento desta pesquisa frente ao estado da arte. Para realizar a análise dos trabalhos selecionados foram definidas cinco categorias, as quais foram atribuídas, conforme as características de cada proposta analisada. Como resultado, foi possível identificar, no universo de trabalho analisado durante essa pesquisa, as principais contribuições do *framework* DaRkaM no contexto dos sistemas de transporte inteligente - ITS.

A escolha das categorias considerou as características principais contidas no *framework* proposto neste trabalho frente aos resultados obtidos a partir do mapeamento sistemático, onde procurou-se identificar o estado da arte para os sistemas de transporte inteligente. O *framework* DaRkaM, conforme já mencionado, tem por fim propor um sistema de gerenciamento de tráfego capaz de atuar em um contexto de fluxo de dados contínuo, utilizando diferentes estratégias de redução de dados na névoa. Diante disso, foram ainda estabelecidas métricas relacionadas ao contexto da redução de dados, redução no uso dos recursos de rede e custos de comunicação. A Tabela 3.1 apresenta os trabalhos identificados frente ao *framework* DaRkaM proposto nesta pesquisa.

A fim de estabelecer uma base comparativa entre os trabalhos analisados, foram selecionadas as seguintes categorias a) Detecção de congestionamento de tráfego, na qual foram analisados trabalhos que abordaram diferentes estratégias de detecção de con-

Tabela 3.1: Matriz de Trabalhos Relacionados

Revisão da Literatura	Deteção de Congestionamento de Tráfego	Redução dos Custos de Comunicação	Redução dos Dados de Tráfego	Fluxo de Dados Contínuo	Intervenções no Cenário Urbano
(CHEN; LI, 2013)	✓	-	-	-	-
(BAUZA; GOZALVEZ; SANCHEZ, 2010)	✓	-	-	-	-
(BAUZA; GOZÁLVEZ, 2013)	✓	-	-	-	-
(HE; ZHANG, 2017)	-	✓	✓	-	-
(SHEN et al., 2019)	-	-	-	✓	-
(GUEDES; CAMPOS, 2016)	-	-	✓	-	✓
(KAUR; KAD, 2016)	-	✓	-	-	✓
(AADIL et al., 2016)	-	✓	✓	-	-
(RAMACHANDRA et al., 2016)	✓	-	-	-	-
(RATH; PATI; PATTANAYAK, 2019)	✓	-	-	-	-
(KUMAR; KUSHWAHA, 2016)	✓	-	-	-	-
(KUMAR; KUSHWAHA., 2019)	✓	-	-	-	-
(TSUYOSHI et al., 2017)	✓	-	-	-	-
(WANG et al., 2019a)	✓	✓	-	-	-
(RANI; SHAW, 2019)	✓	-	-	-	-
(AMELIA; SAPTAWATI, 2014)	✓	-	-	-	-
(ZHANG et al., 2016)	✓	-	-	-	-
(SILVA et al., 2014)	✓	-	✓	-	-
(DING et al., 2015)	✓	-	✓	-	-
(Kerner et al., 2005)	✓	-	✓	✓	-
(TANIZAKI; WOLFSON, 2007)	✓	-	✓	✓	-
(WAHID; RAO; GOEL, 2019)	✓	-	✓	✓	-
(CERQUEIRA et al., 2018)	✓	-	✓	-	-
(SOUZA et al., 2018)	✓	-	✓	-	-
(CHATRAPATHI et al., 2015)	✓	-	✓	-	-
(IRREVALDY; SAPTAWATI, 2017)	✓	-	✓	-	-
(MANJORO; DHAKAR; CHAURASIA, 2016)	✓	-	✓	-	-
(OLIVEIRA; KNISS, 2019)	-	-	✓	-	-
(ZHANG et al., 2020)	-	-	✓	-	-
(ALKHEDER; ALOMAIR, 2021)	✓	-	-	-	-
(MOHANTY et al., 2022)	✓	-	✓	-	-
(SHAH et al., 2022)	✓	✓	✓	-	-
(PEIXOTO et al., 2020)	✓	✓	✓	-	-
(PEIXOTO et al., 2021)	✓	✓	✓	✓	-
(PEIXOTO et al., 2023)	✓	✓	✓	✓	-
Framework DaRkaM	✓	✓	✓	✓	✓

gestonamento, permitindo uma análise frente à estratégia proposta neste trabalho; *b*) Redução dos custos de comunicação, esta categoria representa os trabalhos que abordam preocupações ligadas ao uso dos recursos e seus custos associados; *c*) Redução dos dados de tráfego, que representa o conjunto de trabalhos analisados que implementam estratégias para redução de dados, considerando os níveis de acurácia pós-redução; *d*) Fluxo de dados contínuos, onde foram selecionados trabalhos que abordam o processa-

mento contínuo de dados, especialmente em redes veiculares; *e*) Intervenções no cenário urbano, onde foram classificados os trabalhos que abordam estratégias de intervenção no cenário urbano, em especial, trabalhos que utilizaram de estratégias de roteamento.

Analisando a Tabela 3.1, pode-se observar que muitos trabalhos abordam a temática da detecção de congestionamento, porém, poucos entre os estudos analisados, consideram a quantidade de dados utilizados durante as ações de monitoramento e os seus impactos na disseminação desses conjuntos de dados através de uma rede de comunicação. Nesse contexto, apenas três trabalhos abordam estratégias de redução com foco na detecção de congestionamento em um contexto de fluxo de dados contínuo, entretanto, nenhum desses trabalhos abordam questões relacionadas aos custos de comunicação ou estratégias para tomada de decisão. Adicionalmente, cabe citar que nenhum dos trabalhos analisados se utiliza de técnicas de agrupamento no contexto de fluxo de dados contínuo. Os trabalhos (PEIXOTO et al., 2020) e (PEIXOTO et al., 2021) descrevem estágios anteriores da presente pesquisa que culminaram no desenvolvimento deste *framework*. A Figura 3.1 ilustra as principais contribuições nessas publicações que tiveram o objetivo de submeter à comunidade acadêmica os avanços deste trabalho, conforme novos achados foram identificados. O trabalho em destaque, até o momento da escrita desta tese, encontra-se em situação de avaliação para publicação.

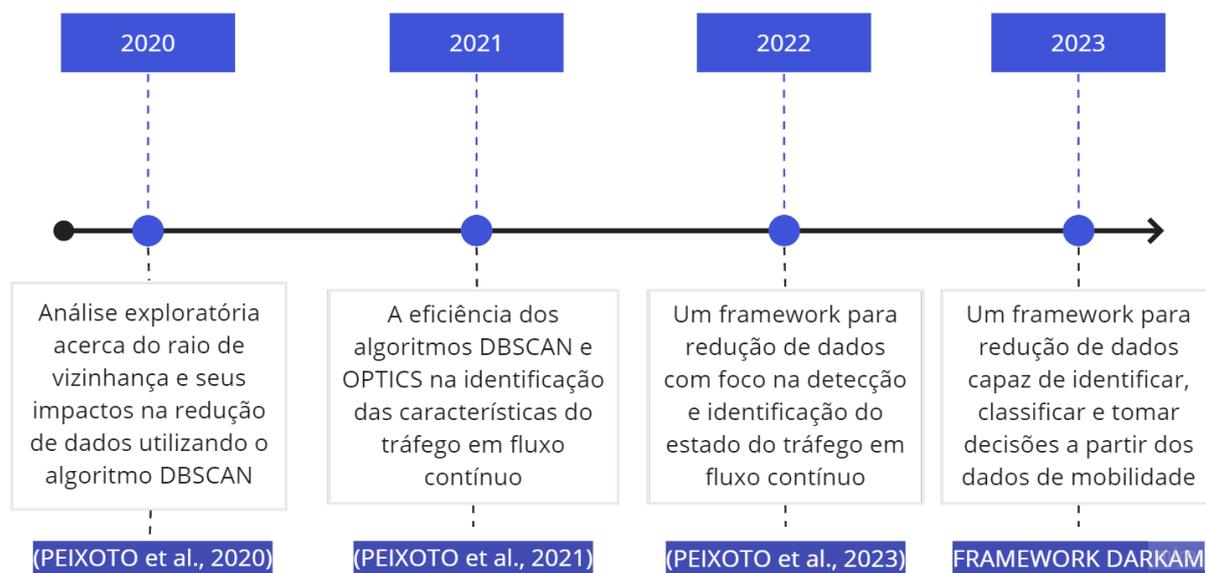


Figura 3.1: Análise temporal dos avanços desta pesquisa

O *Framework DaRkaM*, proposto neste trabalho, traz em seu escopo uma arquitetura capaz de hospedar diferentes técnicas de redução de dados em um contexto de fluxo de dados contínuo na névoa. Por fim, os dados coletados por esta estrutura são processados e servem de insumos para eventuais intervenções no tráfego urbano, com o fim de maximizar a eficiência do tráfego por meio do roteamento de veículos.

### 3.7 CONSIDERAÇÕES FINAIS

Os estudos apresentados ao longo deste capítulo procuram oferecer uma visão ampla acerca do estado da arte, a partir de um conjunto de tópicos reunidos e utilizados na presente proposta para compor o *framework* DaRkaM. Assim, os estudos abordam a temática da computação urbana, em especial no tocante aos sistemas de transporte inteligente, como forma de mitigar os desafios relacionados à eficiência do fluxo de tráfego.

Em um sistema de gerenciamento de tráfego, ações como monitoramento e detecção de congestionamento de tráfego são atividades predominantes para a construção de um sistema de transporte inteligente, capaz de maximizar a eficiência do cenário urbano. Neste contexto, muitas das abordagens apresentadas pelos diferentes autores contribuíram para a confecção desta proposta, seja pela motivação tecnológica, sejam pelas estratégias implícitas nas diferentes abordagens apresentadas.

Ao longo deste capítulo, é possível observar como o paradigma da computação em névoa tem o potencial para maximizar o uso das redes veiculares por meio da extensão dos recursos da nuvem para a borda da rede. Em termos práticos, esta ação aumenta a confiabilidade da latência para sistemas de urgência de tempo real, além de abrir espaço para novos e melhorados sistemas de gerenciamento urbano. Adicionalmente, foi discutido o uso de redes sociais como um meio para aferir o comportamento do tráfego e fornecer informações úteis para a tomada de decisão, uma ação que parece promissora, na medida em que não impele novos custos de equipamento ou comunicação. No entanto, alguns desafios ainda precisam ser superado, como a estratégia de colaboração, segurança nas fontes de dados, integrações, entre outros.

Além disso, foram discutidos estudos utilizando a lógica fuzzy como forma de estabelecer modelos de previsão ou tomada de decisão em redes híbridas. Os estudos demonstram, entre outras coisas, o potencial das redes híbridas e sua eficiência na construção de cenários urbanos amplamente conectados, integrando topologias V2V (*vehicle-to-vehicle*) e V2I (*vehicle-infrastructure*) para formar um sistema de troca de dados mais acurado e capaz de responder às requisições com maior celeridade. Neste cenário, um desafio comum persiste, o volume de dados trafegados.

Outra abordagem discutida refere-se à inserção de algoritmos de agrupamento no contexto da detecção de congestionamento. Tais abordagens têm sido utilizadas em diferentes estudos como forma de aferir a densidade das rodovias. Embora não tenham sido encontradas abordagens relativas ao uso de algoritmos de agrupamento em cenários baseados no fluxo de dados contínuos, outras abordagens foram propostas, entre elas, algoritmos baseados em amostragem simples para seleção e redução de dados, alguns deles utilizados na presente proposta. Por fim, analisa-se neste capítulo as questões relativas ao aumento do volume de dados alavancado pelas redes veiculares e a relevância da redução de dados como forma de contribuição para a adoção ampla das tecnologias de monitoramento e sistemas de transporte inteligente nos cenários urbanos.

## DATA REDUCTION FRAMEWORK FOR TRAFFIC MANAGEMENT (DaRkaM)

Este capítulo apresenta o framework DaRkaM acrônimo em inglês para (*Data Reduction Framework for Traffic Management*). Ao longo das próximas seções, serão discutidos e apresentados os ambientes, aspectos metodológicos, estrutura e arquitetura do *framework* proposto neste trabalho.

### 4.1 ASPECTOS CONCEITUAIS

Os sistemas de transporte inteligente integram uma parte fundamental da computação urbana. Ao lidar com os desafios do tráfego, esses sistemas contribuem para mitigar outros desafios do espaço urbano e seus impactos podem ser observados não só pela atuação preventiva na formação de congestionamentos, cujos reflexos ambientais são observados pela redução de CO<sub>2</sub>, melhoria na qualidade do ar, reduções no consumo combustível e energia, mas também pela maximização da eficiência do fluxo de tráfego, transportes públicos e na própria experiência ao dirigir, aspectos que, em última análise, se traduzem na melhoria da qualidade de vida do cidadão.

Com isso em mente, foi desenvolvido ao longo dessa pesquisa o *framework* DaRkaM, um sistema de transporte inteligente que se utiliza do paradigma da computação em névoa para identificar e mitigar os desafios relacionados ao fluxo de tráfego nas rodovias. O *framework* atua no monitoramento e gestão do tráfego por meio de um modelo de processamento em camadas que atua de forma orquestrada para oferecer alternativas para maximizar a eficiência do tráfego nos centros urbanos.

A presente pesquisa abarca em seu escopo um modelo de redução de dados capaz de hospedar diferentes abordagens de seleção e descarte de dados. O módulo de redução de dados proposto neste *framework* tem por finalidade reduzir os custos de comunicação durante as ações de monitoramento e gerenciamento de tráfego. Com isso, espera-se reduzir a relação proporcional entre o aumento da quantidade de dados ou veículos e o consequente aumento nos custos de comunicação. Nas próximas seções são discutidos os detalhes da implementação em cada uma das camadas deste modelo.

### 4.2 ARQUITETURA DARKAM

Esta seção apresenta os elementos que compõem o framework DaRkaM proposto neste trabalho. A Figura 4.1 ilustra detalhes do modelo arquitetural proposto.

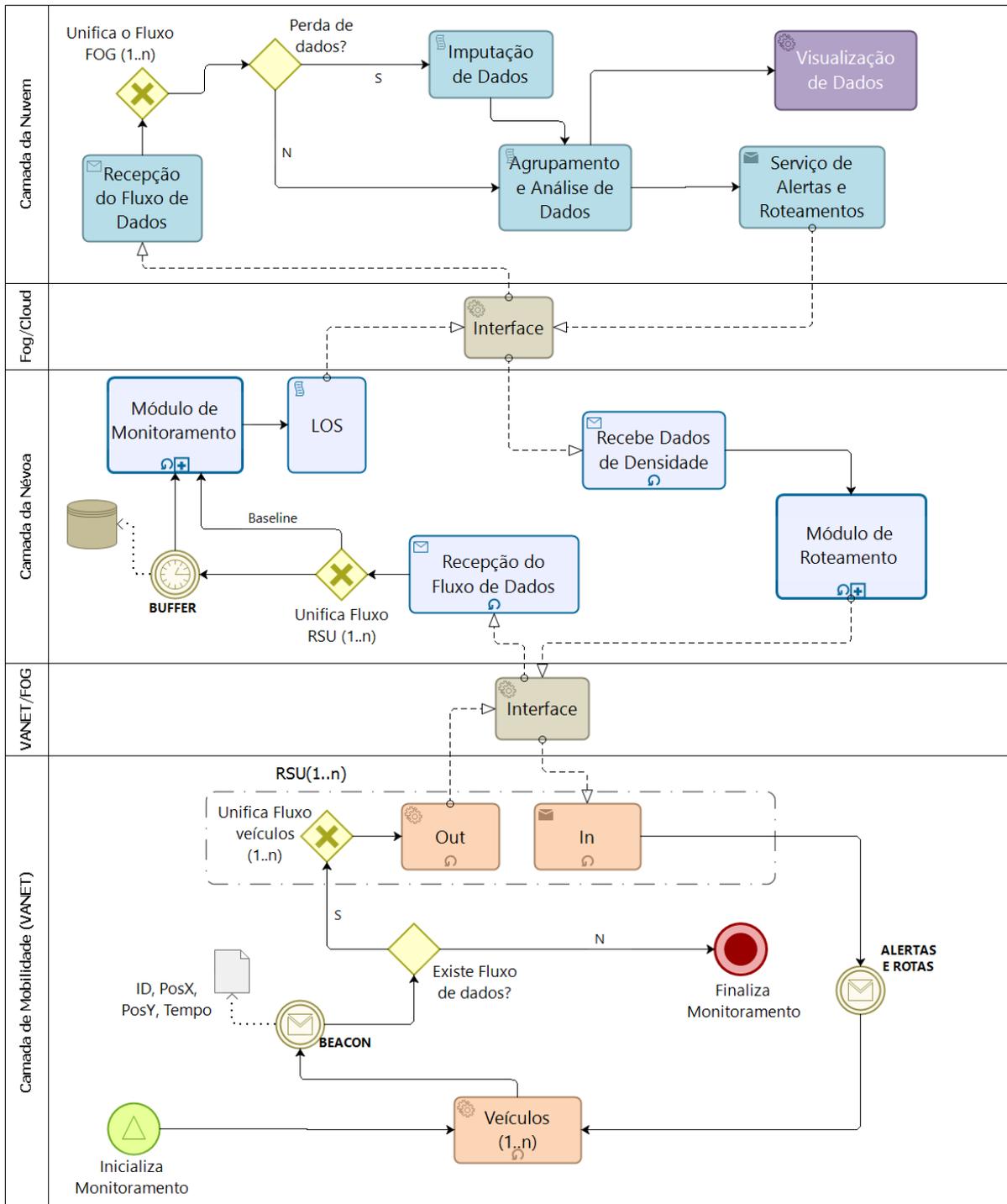


Figura 4.1: Elementos Arquiteturais que Compõem o Framework DaRkaM

Conforme mencionado anteriormente, o modelo proposto é formado por três camadas que atuam de forma integrada para formar um *framework* capaz de lidar com desafios como redução da latência, redução do volume de dados e as consequências destas reduções nos custos de comunicação. Adicionalmente, vale destacar que toda a arquitetura presume a existência de um fluxo de dados contínuo que tem origem na camada de mobilidade e segue em direção às camadas adjacentes do modelo.

Nesse contexto, o fluxo de dados contínuo produzido pelo ambiente de mobilidade é enviado para as próximas camadas por meio de interfaces de comunicação. Essas interfaces garantem, por meio de um cabeçalho comum, o desacoplamento entre as camadas, tornando-se autocontidas, evitando que o impacto de ajustes realizados em uma camada intervenha no funcionamento de outra.

Os dados obtidos do ambiente de mobilidade são processados pela névoa. No núcleo desta camada estão *a)* o módulo de monitoramento, que atua no tratamento do fluxo de dados recebido da camada de mobilidade e, *b)* o módulo de roteamento, o qual tem a função de propagar as decisões tomadas pela camada de nuvem no ambiente de mobilidade. A tomada de decisão é então enviada ao ambiente de mobilidade por meio da mesma interface comum. A Figura 4.1 ilustra essa arquitetura.

As camadas descritas nesta arquitetura podem ser definidas ainda em um fluxo de ações onde cada veículo  $n_i \in N$  envia periodicamente uma mensagem *beacon* ( $B_j$ ) para a RSU mais próxima. O fluxo de atividades desta arquitetura pode ser descrito da seguinte forma:

- **Etapa 1;**  $B$  representa o *beacon*, assim,  $B_j$  é enviado pelos veículos  $n_i$  para uma  $RSU_i$  dentro do raio de cobertura.  $B_j$  carrega em sua estrutura uma tupla contendo informações como: posição do veículo ( $P_{x,y}^i$ ) e identificação ( $id_i$ ). Por meio de uma conexão sem fio, as informações são encaminhadas da  $RSU_i$  para um nó da névoa  $Fog_i$  em um fluxo contínuo de dados.
- **Etapa 2.** Cada nó  $Fog_i$  pode lidar com um grupo de RSUs. O processamento da névoa tem por função reduzir o volume de dados trafegados pela rede, aliviando os links LTE-5G entre a névoa e a nuvem. Esta ação contribui para mitigar o problema de crescimento de dados na fonte, além de atuar na redução da latência, devido à proximidade dos veículos com os recursos de borda.
- **Etapa 3.** Dados processados na névoa são enviados para a nuvem. Nesse contexto, a nuvem atua recebendo dados em fluxo dos diferentes nós  $Fog_i$  para montar um panorama sob o qual o processo de tomada de decisão irá ocorrer.
- **Etapa 4** Decisões tomadas na camada da nuvem são propagadas até o ambiente de mobilidade, utilizando o mesmo padrão de interfaces. Essas decisões envolvem aspectos como a análise da situação do tráfego com base na observação da acurácia, utilizando a métrica LOS e sugestões dinâmicas para rotas alternativas. Detalhes arquiteturais serão apresentados nas seções a seguir.

### 4.3 FLUXO DE DADOS ENTRE AS CAMADAS

Na presente pesquisa, assume-se que cada dispositivo está emitindo um fluxo de dados contínuo que pode ser recepcionado por diferentes unidades de processamento RSUs. Conforme estes conjuntos de dados são recebidos, são encaminhados para a camada subsequente, na qual os dados são reunidos formando um fluxo de dados unificado. Pode-se descrever um fluxo de dados ( $S$ ) como uma sequência de tuplas ilimitadas geradas continuamente ao longo do tempo. Assim,  $S = \{(S_1, \tau_1), (S_2, \tau_2), \dots\}$ , onde  $S$  é uma sequência infinita de elementos multidimensionais, ou seja, carregam em sua estrutura diferentes informações de tráfego de veículos. O fator tempo é dado por  $\tau \in R_+^* = \{x \in R | x \geq 0\}$ . Conforme o tempo limite é alcançado, o conjunto de dados  $S$  é submetido aos processos de agrupamento e redução de dados de tráfego. Concomitantemente, novos dados de tráfego de entrada se tornam disponíveis em janelas temporais subsequentes.

O recurso das janelas temporais consiste no particionamento dos elementos em uma série de segmentos  $S' = \{p_1, p_2, \dots, p_{|S'|}\}$ . Conforme já mencionado, este subconjunto deve manter-se alinhado ao tamanho de janela  $\gamma$ . Durante o processamento, pode-se observar que a janela  $\tau$ -th é reformulada para um *micro batch* facilitando o processamento dos dados pelas camadas adjacentes, uma vez que apenas o  $S$  mais recente com um tamanho  $\gamma$  fixo é mantido em  $D$  na janela deslizante. Os elementos na janela deslizante seguem um método de escalonamento *first in, first out* (FIFO), que considera elementos do período atual ( $\tau_i$ ) até um período no passado ( $\tau_{i-j}$ ), onde  $i$  é o início e  $j$  o fim do intervalo de tempo  $\Delta t$ . A abstração do processamento de fluxo de dados contínuos realizado pelo framework desenvolvido nessa pesquisa é ilustrado na Figura 4.2, onde  $n_i$  representa a identificação de cada veículo no raio de cobertura de uma RSU.

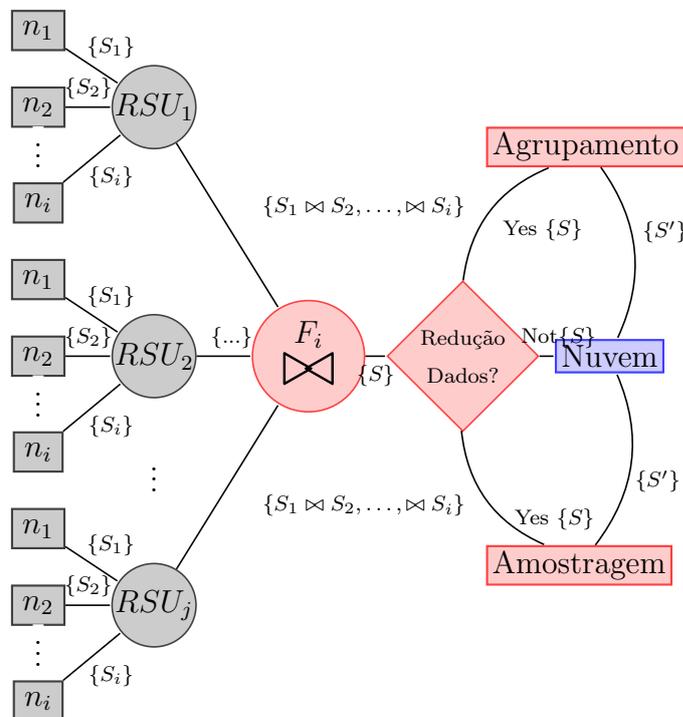


Figura 4.2: Processamento do Fluxo de Dados Contínuo no *Framework* DaRkaM

Os elementos cinza na Figura 4.2, ou seja, o conjunto de veículos  $\{n_1, n_2, \dots, n_i\}$  e o conjunto de RSUs  $\{rsu_1, rsu_2, \dots, rsu_j\}$ , fazem parte da Camada VANET. Cada veículo  $n_i$  envia um fluxo de dados de tráfego  $\{S_i\}$  para  $RSU_j$  em sua área de cobertura. Todos os fluxos de dados de tráfego são unidos por meio de uma operação  $\bowtie$  e o fluxo resultante  $\{S_j\}$  é encaminhado para o dispositivo de computação em névoa mais próximo. Os elementos em vermelho na Figura 4.2 indicam os métodos empregados pelo framework Darkam na camada da névoa. A primeira etapa do processamento nesta camada é reunir todos os dados de tráfego oriundos das diferentes RSUs por meio de uma operação  $\bowtie$ . Embora seja possível encaminhar esse conjunto de dados  $\{S\}$  diretamente para a nuvem, dois tipos de redução de tráfego de dados são implementados: (i) com base em amostragem e (ii) com base em agrupamento. Estes processos de redução de dados propostos nessa pesquisa serão detalhados nesse capítulo.

#### 4.4 SIMULADORES UTILIZADOS NA CAMADA DE MOBILIDADE

Conforme já mencionado, a natureza distribuída das redes veiculares dificulta a prática de estudos experimentais em cenários reais, assim, torna-se necessária uma alternativa para a realização de experimentos de modo a estabelecer um entendimento sólido acerca de como seria a aplicação de certas tecnologias, métodos e técnicas nestes cenários. Os simuladores representam uma possível solução para este desafio.

Assim, propõe-se nessa pesquisa, o uso de três simuladores para reproduzir os aspectos fundamentais da mobilidade de tráfego, são eles: (SUMO, 2019), escolhido pela sua capacidade de simular o tráfego rodoviário contínuo; (OMNET++, 2019) responsável por fornecer funcionalidades relativas à simulação de redes e conectividade; e por fim (VEINS, 2019), responsável por integrar as definições de mobilidade gerenciadas pelo simulador SUMO com os aspectos de conectividade gerenciados pelo OMNET++. Estes simuladores deverão operar de forma ordenada para gerar um fluxo contínuo de dados de tráfego, que serão capturados, inicialmente, pela infraestrutura mobilidade, para então serem direcionados às diferentes camadas do framework.

Para fazer isso, o simulador SUMO foi configurado com um mapa da cidade de Colônia, Alemanha, extraído do *OpenStreetMap*. Este mapa foi carregado dinamicamente com diferentes densidades veiculares com o fim de simular diferentes situações de tráfego. Os recursos de conectividade foram configurados no OMNET++ onde o protocolo IEEE 802.11p foi definido como protocolo de comunicação padrão para os experimentos. Por fim, uma RSU foi posicionada no cenário para permitir a comunicação entre os veículos e a infraestrutura (V2I) da rodovia. Cabe citar ainda, que toda a infraestrutura de comunicação entre o SUMO e o OMNET++ foi realizada por meio do simulador VEINS. Seus métodos e padronização para troca de mensagens permitem o desenvolvimento de funções orientadas e eventos sob os quais foi implementada toda a infraestrutura de mobilidade utilizada nessa pesquisa.

## 4.5 CAMADA DE MOBILIDADE - VANET

Um sistema de transporte inteligente é estruturado, essencialmente, sob o ambiente de mobilidade urbana e envolve características como arquitetura e a capacidade de comunicação entre os elementos dispostos em um certo cenário. Na presente pesquisa, a camada de mobilidade é responsável por produzir o fluxo contínuo de dados contendo informações essenciais sobre a identificação do estado do tráfego ao longo do tempo.

Os cenários de mobilidade podem ser definidos como um conjunto de grafos ponderados, baseados nas informações das rodovias, sob as quais pode-se realizar análises espaciais e/ou temporais. Uma formalização deste problema pode ser observada a seguir, na Definição 12.

**Definição 12.** *Seja uma VANET modelada como um grafo direcionado e ponderado  $G = (V, E)$ , onde  $V = \{v_1, v_2, \dots, v_{|V|}\}$  é o conjunto de interseções e  $E = \{e_1, e_2, \dots, e_{|E|}\}$  é o conjunto de segmentos de estrada conectando as interseções. Supondo que esteja  $N = \{n_1, n_2, \dots, n_{|N|}\}$  representando os veículos na estrada, quando um veículo  $n_i$ , onde  $i$  representa o  $i^{\text{th}}$  veículo  $\in N$ , está dirigindo de uma interseção de origem  $v_o$  para uma interseção de destino  $v_p$ , um conjunto ordenado de segmentos de estradas pode ser definido então como  $U_{o,p} = \{e_1 e_2, \dots, e_e\}$ , onde  $|e|$  é o número de segmentos na rota.*

Durante o fluxo de tráfego, esses *beacons* são enviados em alta frequência pela rede veicular e tais envios independem da recepção dos mesmos por outros equipamentos da rede. A frequência com que os *beacons* são emitidos no cenário da rede veicular justifica-se no fato de que veículos, em geral, não sabem se estão sob uma área de cobertura e nem se os seus *beacons* estão sendo recuperados por uma *Road Side Unit* (RSU). Assim, torna-se imprescindível que novos elementos sejam reconhecidos o mais rápido possível. Entretanto, por outro lado, envios em alta frequência podem representar um risco de sobrecarga da rede. A abordagem baseada em redução de dados desenvolvida durante essa pesquisa procura mitigar esses efeitos.

Neste trabalho, as *Road Side Units* (RSU) são responsáveis por encaminhar os fluxos de dados recebidos pelos diferentes veículos. Adicionalmente, cada unidade de processamento está conectada a uma interface de entrada e saída que controla o fluxo de dados que chega ou sai da camada de mobilidade. Os diferentes fluxos de dados recebidos na RSU são emitidos para uma interface que é responsável por conduzir esses conjuntos às camadas adjacentes do modelo.

Analogamente, as decisões tomadas nas camadas superiores são recepcionadas por esta mesma interface e emitidas a uma RSU posicionada na camada de mobilidade. Esta mensagem pode conter desde um alerta acerca da formação de congestionamentos até a sugestão de rotas, a fim de maximizar o fluxo de tráfego nas rodovias adjacentes. A camada de mobilidade permanece realizando o envio de dados em fluxo até que não existam mais dados a serem enviados, neste ponto o monitoramento é finalizado.

Essa abordagem apoia-se no trabalho de (SCHMIDT et al., 2010), onde assume-se que o envio de mensagens pelos veículos seja realizado na frequência de 1Hz, esta média se refere a quantidade de *beacons* emitidos por segundo a partir de um único carro para

uma certa RSU. Esta medida pode ser caracterizada como o mínimo possível para manter o monitoramento eficiente na formação de pontos de retenção ou congestionamentos. Adicionalmente, esta proposta considerou que existem dois padrões de comunicação V2I para lidar com o fluxo contínuo de dados de tráfego, conforme ilustrado na Figura 4.3.

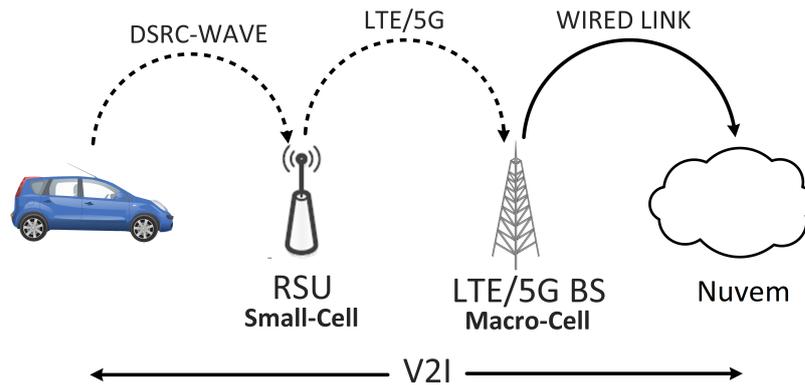


Figura 4.3: Modelo de Comunicação Veículo / Infraestrutura

Primeiro, o veículo envia os dados de tráfego diretamente para a RSU, que é encaminhado para a *Base Station* (BS) e posteriormente para uma nuvem ou para um servidor centralizado. A RSU recebe os dados de tráfego por meio do protocolo IEEE 802.11p que atua como uma célula na arquitetura 5G. Um cabeçalho comum foi definido como forma de padronizar a troca de informações entre as camadas.

Cada RSU se comunica com os veículos por meio de uma conexão *Wireless Access in Vehicular Environments* (WAVE) para coletar informações de tráfego como horário, velocidade e localização dos veículos. Os dados de tráfego adquiridos são encaminhados para a camada de névoa. O protocolo IEEE 802.11p, tecnologia de rádio de comunicação de curto alcance (WU et al., 2013), foi utilizado como forma de viabilizar a trocar dados entre veículos e as unidades de processamento RSU. Essa arquitetura contribui, não só para a comunicação entre os elementos do espaço urbano, como também mantém um importante alinhamento com o desenvolvimento de ecossistemas baseados na Internet das Coisas (IoT).

## 4.6 CAMADA DE NÉVOA

Um dos principais objetivos do *framework* DaRkaM é minimizar o custo de comunicação de dados de tráfego sobre os *links* de rede LTE/5G. Neste contexto, a camada de névoa é responsável por lidar com os processos de redução dos dados de tráfego, além de mitigar problemas de latência e sobrecarga da rede. Esta camada atua como receptora de todo o fluxo de dados do ambiente de mobilidade e fornece estratégias de redução com base em elementos como densidade e distribuição dos veículos ao longo das rodovias. Uma formalização deste problema é apresentada na Definição 13.

**Definição 13.** *Seja um veículo  $N$  modelado como  $N = (n_1, n_2, \dots, n_i)$ , enviando periodicamente um conjunto de dados de tráfego ( $B = id, P^{x,y}$ ) para uma RSU mais próxima, onde  $id$  é o número de identificação do veículo e  $P^{x,y}$  fornece a localização do veículo  $n_i$ , sendo  $x$  a latitude e  $y$  a longitude. Supondo que a velocidade  $s(i)$  de cada veículo  $n_i$  é obtida pela diferença entre dois pontos de posição consecutivos ( $P_{x_i, y_i} - P_{x_{i-1}, y_{i-1}}$ ). Portanto, a quantidade total de fluxo de dados de tráfego  $\sum_{i=0}^{|N|} (\sum_{j=0}^{|B|} B_j^i)$  é agrupada em  $R = (\sum_{i=0}^{|N|} (\sum_{j=0}^{|B|} B_j^i)) / \Omega \mid (\forall n \in N : \Omega \geq 1)$ , onde  $\Omega$  é o fator de redução de dados e  $R$  é a redução de tráfego alcançada.*

Na presente pesquisa, uma interface entre as camadas é responsável por realizar a distribuição dos dados na camada de névoa. Assim, dados obtidos da camada de mobilidade são enviados em fluxo contínuo para uma certa RSU, e esta, por sua vez, se encarregará da unificação e reencaminhamento destes dados para a névoa.

Esta arquitetura garante o desacoplamento entre as camadas, permitindo que as mensagens sejam trocadas sem que exista a necessidade de uma camada conhecer detalhes do processamento de outra. Entre os benefícios deste modelo arquitetural estão o encapsulamento ou modularização da aplicação, além da redução dos riscos relacionados à manutenção destes componentes (ALLEN; KHOSHGOFTAAR; CHEN, 2001).

Os fluxos de dados recebidos na névoa podem ser provenientes de diferentes RSUs. Deste modo, um mecanismo de unificação destes conjuntos de origens difusas é necessário. Diante disso, os dados recebidos da *interface* são primeiramente organizados em um único conjunto de dados para que então sejam submetidos ao processamento de névoa. O tratamento do fluxo de dados ocorre por meio de uma janela temporal. Essa janela atua como um mecanismo para estabelecer limites flexíveis de processamento em um fluxo ilimitado de dados.

A janela temporal tem por fim garantir a seleção de um conjunto finito e variável de tuplas que serão acumuladas durante um certo intervalo de tempo, como explicado em (TOLPIN, 2016) e (PATROUMPAS; SELLIS, 2006). Na presente proposta, a janela temporal tem a função de criar um *buffer* de dados que serão sequencialmente submetidos a processamento. Em termos práticos, os valores temporais das tuplas contidas no fluxo de dados são encaminhados para processamento em um intervalo de tempo pré-especificado.

O uso de Janelas temporais tem potencial para produzir uma resposta aproximada a uma consulta no fluxo de dados, permitindo que os algoritmos executados nessa camada atuem sobre as partes mais recentes do fluxo de dados ao invés de analisar todo o histórico a cada iteração. Portanto, a janela deslizante reúne os dados do fluxo em um intervalo de tempo ( $\Delta t$ ) e quando o *tempo\_decorrido* > *tempo\_limite\_janela*, os dados acumulados formam o subconjunto que será submetido ao processamento pelo módulo de monitoramento.

#### 4.6.1 Módulo de Monitoramento

O módulo de monitoramento foi projetado com o fim de hospedar diferentes abordagens com foco na análise, seleção e a redução de dados na borda da rede. Com uma

arquitetura de programação em camadas, este modelo permite que novas abordagens de monitoramento e redução de dados possam ser incorporadas ao núcleo do módulo de monitoramento sem impactar nas estruturas já implementadas. Esse modelo oferece maior versatilidade na medida em que promove a possibilidade de alternar as estratégias, conforme eventuais requisitos do ambiente, além de permitir a realização de medições da acurácia e dos níveis de redução de dados alcançados.

Adicionalmente, o *framework* DaRkaM incorpora na sua estrutura um conjunto de algoritmos que serão detalhados a seguir, estes algoritmos têm a função de operacionalizar as estratégias de monitoramento e redução de dados. As saídas do módulo de monitoramento são essencialmente os subconjuntos de dados reduzidos, os quais servirão de entrada para o processo da classificação do fluxo de tráfego. Este processo utiliza o padrão de classificação conhecido como nível de serviço (*Level of Service* - LOS) para rotular o tráfego nas diferentes rodovias monitoradas.

Os algoritmos hospedados no módulo de monitoramento foram escolhidos com base na literatura. Neste sentido, foram selecionadas abordagens cujas diferenças estruturais e de processamento pudessem trazer uma visão ampla acerca dos seus respectivos comportamentos. Assim, foram definidos, inicialmente, oito algoritmos dos quais três são baseados no conceito de amostragem simplificada, sendo eles **1to2**, **Random** e **Limite**. A rigor, estes algoritmos atuam selecionando os dados durante a leitura do fluxo de dados contínuo; Quatro algoritmos baseados em agrupamento, sendo eles **DBSCAN**, **X-Means**, **OPTICS** e **HDBSCAN**. Adicionalmente, um algoritmo **Baseline** também foi implementado para fins de medições.

Algoritmos de agrupamento, em geral, não estão plenamente adaptados para atuar em um contexto de fluxo de dados contínuo, assim, os algoritmos **DBSCAN**, **X-Means**, **OPTICS** e **HDBSCAN** foram adaptados de modo a viabilizar este modelo de processamento. A adaptação destes algoritmos consiste na definição de uma janela temporal capaz de criar um *buffer* que compreende o estado geral do ambiente de mobilidade em um certo momento. Estes dados são então submetidos ao processamento pelos respectivos algoritmos e seus resultados são utilizados nas estratégias de redução e análise de dados. Sob estes subconjuntos de dados, já reduzidos, são tomadas as decisões acerca das eventuais intervenções no espaço urbano. A seguir, são analisados os algoritmos utilizados neste trabalho. Um resumo acerca do funcionamento dos algoritmos utilizados nessa pesquisa é apresentado a seguir.

#### 4.6.1.1 Algoritmo Baseline

Trata de uma abordagem que não utiliza qualquer método de redução. Sua função é estabelecer uma base comparativa confiável, capaz de oferecer um parâmetro de controle que possa ser utilizado como medida de acurácia para as outras abordagens implementadas durante a presente pesquisa (SOUZA; VILLAS, 2016; SOUZA et al., 2016).

Assim como os demais algoritmos implementados no módulo de monitoramento, o processamento do *baseline* ocorre na camada da névoa e consiste, essencialmente, na recuperação e processamento do fluxo de dados contínuo recebido a partir camada de mobilidade. Dados oriundos das unidades de processamento são submetidos a uma janela

temporal e processados na íntegra, sem que haja a aplicação de nenhuma técnica de redução. Assim, uma vez que o processamento dos dados ocorre integralmente, tende-se a acreditar que seus resultados podem servir como um parâmetro confiável para a aferição das medidas de acurácia implementadas nas diferentes estratégias de redução implementadas neste trabalho. O Algoritmo 1 ilustra o funcionamento desta abordagem.

---

**Algorithm 1:** Baseline
 

---

**Require:** Conjunto de pontos obtidos no fluxo de dados *Data* contendo todas as informações posicionais do veículo no intervalo de tempo  $\Delta t$

**while** ( $\Delta t == \text{LenghtWindow}$ ) **do**  
      $buffer \leftarrow D$   
**end while**  
 $Data \leftarrow buffer$   
*Inicia leitura dos dados*  
**while** ( $!Data \text{ IsEmpty}$ ) **do**  
      $beacon \leftarrow Data$   
**end while**  
**return**  $beacon$

---

Para permitir o ingresso do algoritmo *baseline* no contexto do módulo de monitoramento foi necessário realizar uma adaptação com o fim de garantir o seu funcionamento em um ambiente de fluxo de dados contínuo. Para realizar este ajuste, o algoritmo se inicia com o monitoramento dos dados recebidos pela camada de monitoramento. Este processamento ocorre durante um certo intervalo de tempo  $\delta t.$ , ao qual convencionou-se nomear de janela temporal. Conforme os dados são recebidos, um *buffer* é criado para reunir os dados que serão processados na próxima fase. Uma vez finalizada a janela temporal, este *buffer* é submetido ao processamento. Como o algoritmo *baseline* tem por função utilizar os dados integralmente, o processamento consiste essencialmente em descapsular os *beacons* contidos no conjunto de dados e submeter às próximas fases do *framework* DaRkaM.

#### 4.6.1.2 1-TO-2

Proposto por (TANIZAKI; WOLFSON, 2007), essa abordagem baseia-se no conceito de amostragem simples. O algoritmo é executado sem considerar os atributos relativos aos dados de tráfego, como velocidade, fluxo de tráfego ou densidade. Seu processo de seleção consiste no monitoramento dos dados, os quais são enviados a partir de uma formulação lógica na qual a cada um *beacon* enviado, dois são descartados. Assim, o algoritmo se utiliza de uma taxa de [1 : 2] para realizar o processo de seleção de dados. Para incorporar este modelo de processamento no módulo de monitoramento na névoa, foi necessário realizar a sua adaptação para um contexto de fluxo de dados contínuo. Similarmente a abordagem anterior, os dados recebidos pela camada de mobilidade são submetidos a

uma janela temporal que tem por função reunir os dados durante um intervalo de tempo  $\delta t$ . O conjunto de dados resultante é submetido ao processo de redução de dados por meio da aplicação do algoritmo 1TO2. Uma descrição detalhada do seu funcionamento é apresentada no Algoritmo 2.

---

**Algorithm 2:** 1-TO-2
 

---

**Require:** Conjunto de pontos obtidos no fluxo de dados *Data* contendo todas as informações posicionais do veículo no intervalo de tempo  $\Delta t$

```

pos = 0
while ( $\Delta t == \text{LenghtWindow}$ ) do
  buffer  $\leftarrow D$ 
end while
Data  $\leftarrow$  buffer
Inicia leitura dos dados
while (!Data IsEmpty) do
  if (pos == 2) then
    beacon  $\leftarrow$  Data
    pos = 0
  end if
  pos+ = 1
end while
return beacon

```

---

Analisando o algoritmo, pode-se observar que o processo se inicia com a criação de um contador denominado *pos*, que será utilizado posteriormente na aplicação da lógica de seleção por amostragem simples. Conforme já mencionado, os dados são agrupados em uma janela temporal que realiza a reunião dos dados recebidos pela camada de mobilidade. Quando o limite da janela é alcançado, os dados são encaminhados para processamento e, neste ponto, ocorre a redução de dados na névoa. O processamento consiste no controle de uma condição de seleção que deve ser atendida a cada dois *beacons* descartados. Quando este subconjunto é inteiramente analisado, novas janelas são liberadas, gerando um fluxo de análise contínuo que ocorrerá enquanto existirem dados a serem analisados.

#### 4.6.1.3 Random

O algoritmo Random foi proposto em (TANIZAKI; WOLFSON, 2007) e sua abordagem preconiza a utilização de um modelo de seleção aleatório responsável por decidir se um *beacon* será ou não selecionado pelo algoritmo. Para fazer isso, o algoritmo Random se utiliza de uma função de atualização aleatória na qual os dados são selecionados apenas quando uma certa probabilidade  $p > 1$  é alcançada. Assim como na integração dos demais algoritmos, esta abordagem também demandou a realização de adaptações que permitissem a incorporação dessa proposta a um contexto de fluxo de dados contínuo.

Nesse contexto, dados resultantes de uma janela temporal são submetidos a um processo de seleção aleatória na qual um subconjunto reduzido será retornado. Uma descrição detalhada do seu funcionamento é apresentada no Algoritmo 3.

---

**Algorithm 3:** Random
 

---

```

Require: Conjunto de pontos obtidos no fluxo de dados Data contendo todas as
informações posicionais do veículo
no intervalo de tempo  $\Delta t$ 
while ( $\Delta t == \text{LenghtWindow}$ ) do
    buffer  $\leftarrow D$ 
end while
Data  $\leftarrow \text{buffer}$ 
Inicia leitura dos dados
while ( $\text{rand}[0..1] == \text{True}$ ) do
    beacon  $\leftarrow \text{Data}$ 
end while
return beacon
  
```

---

Similarmente aos demais algoritmos já analisados, baseados em amostragem simples, essa abordagem consiste na definição de uma janela temporal é que responsável por reunir os dados ao longo de um certo intervalo de tempo  $\delta t$ . Quando o tamanho da janela é alcançado, os dados são submetidos ao processo de redução. Ao contrário das abordagens já discutidas, não há como mensurar de forma objetiva a taxa de redução de dados a cada iteração, uma vez que a aleatoriedade na decisão sobre a seleção pode conduzir a taxas de redução maiores ou menores nas diferentes execuções. O processo de redução ocorre por meio de uma função de aleatoriedade  $Rand()$ , na qual uma decisão é tomada com base em um retorno binário que indica se o *beacon* será ou não selecionado no subconjunto final.

#### 4.6.1.4 Limite

O algoritmo Limite foi proposto primeiro por (WAHID; RAO; GOEL, 2019) e sua abordagem consiste na definição de um *threshold* que indica o limite mínimo para que um dado seja selecionado. No contexto da mobilidade urbana, o limite foi estabelecido com base na velocidade dos veículos, assim, *beacons* recebidos pela camada de mobilidade são analisados e a velocidade de cada veículo é aferida frente a esse *threshold*  $\epsilon$ . Cabe citar ainda que este limite é definido como um parâmetro do algoritmo, de modo que a decisão acerca de um melhor limite passa pelo entendimento claro do cenário que se deseja analisar. Nesse contexto, as variações constantes da densidade urbana tornam essa tarefa nada trivial. Adicionalmente, assim como nas abordagens anteriores, foi realizada uma adaptação para garantir o seu funcionamento em um cenário de fluxo de dados contínuo. Assim, foi definida uma janela temporal sob a qual o conjunto de dados resultante é analisado. O processamento dos dados ocorre com base na aferição da velocidade carregada

em cada *beacon* analisado. Velocidades abaixo do *threshold* são descartadas formando um subconjunto original cujo valor mínimo de velocidade é estabelecido. Uma descrição detalhada do algoritmo Limite é apresentada no Algoritmo 4.

---

**Algorithm 4:** Limite

---

**Require:** Conjunto de pontos obtidos no fluxo de dados *Data* contendo todas as informações posicionais do veículo no intervalo de tempo  $\Delta t$

**Require:** um parâmetro  $p$  que indica o limite mínimo para o *threshold*.

*threshold* =  $p$

**while** ( $\Delta t == \text{LenghtWindow}$ ) **do**

*buffer*  $\leftarrow D$

**end while**

*Data*  $\leftarrow$  *buffer*

*Inicia leitura dos dados*

**while** (*Data.speed*  $\geq$  *threshold*) **do**

*beacon*  $\leftarrow$  *Data*

**end while**

**return** *beacon*

---

Analisando o Algoritmo 4, pode-se observar a necessidade e incorporar um parâmetro  $p$  que deverá ser utilizado para estabelecer o limite mínimo relativo à velocidade dos veículos. Uma vez definido um parâmetro, a janela temporal é inicializada. Quando o limite da janela é alcançado, os dados são submetidos ao processamento e redução de dados. A redução neste caso ocorre com base no *threshold*, na medida em que os dados abaixo deste limite são sumariamente descartados. Assim, o processamento consiste em iterar sob todo o conjunto de dados resultante da janela temporal, analisando as diferentes velocidades extraídas dos *beacons* recebidos pela camada de mobilidade. O conjunto de dados resultante é então enviado para as camadas adjacentes do modelo. Cabe citar que o estabelecimento do limite assertivo é fator preponderante não só para alcançar altas taxas de redução de dados, mas principalmente para que estas reduções não ofereçam significativa perda de acurácia. Esta seleção pode ser um desafio no uso dessa abordagem.

#### 4.6.1.5 DBSCAN

Proposto por (ESTER et al., 1996), o algoritmo DBSCAN é, conforme já mencionado, um método não paramétrico que, dado um conjunto de veículos no mapa, agrupa veículos próximos uns dos outros. DBSCAN utiliza-se de dois parâmetros principais, *i*)  $\epsilon$  representando a distância máxima entre dois veículos, ou seja, a distância da vizinhança e *ii*) *minPts* descrevendo o número mínimo de veículos para criar um *cluster*. Assim como nas abordagens baseadas em amostragem, apresentadas anteriormente, propõe-se neste trabalho, uma adaptação técnica e conceitual do algoritmo DBSCAN, uma descrição detalhada do seu funcionamento é ilustrada pelo Algoritmo 5.

---

**Algorithm 5:** DBSCAN Adaptado: Redução de Dados em fluxo contínuo
 

---

**Require:** Conjunto de pontos obtidos no fluxo de dados ( $D$ ) por  $\Delta t$

**Require:**  $minPts$ ,  $\epsilon$ ,  $percent$

**Ensure:** agrupamentos ( $k$  e  $k'$ ) para cada  $\Delta t$

**while**  $\Delta t == LengthWindow$  **do**

$buffer \leftarrow D$

**end while**

$D \leftarrow buffer$

*Inicializa o conjunto de clusters  $k = 0$*

**for all** novo ponto  $P \in D$  **do**

**if**  $P$  is visited **then**

$P_i = P_{i+1}$

**else**

$P = visited$

**end if**

$NhoodSet = \text{all } P \in \epsilon.neighborhood$

**if**  $|NhoodSet| < minPts$  **then**

$P = noise$  *#classifica como ruído*

**else**

$k_i = k_{i+1}$

$k = P$

**for all**  $P' \in NhoodSet$  **do**

**if**  $P'$  is not visited **then**

$P' = visited$

$NhoodSet' = \text{all } P' \in \epsilon.neighborhood$

**if**  $|NhoodSet'| \geq minPts$  **then**

$NhoodSet = NhoodSet'$

**end if**

**end if**

**if**  $P' \notin k \leftarrow (k = 0, \dots, k = n)$  **then**

$k = P'$  *#O ruído de um grupo pode ser absorvido por outro*

**end if**

**end for**

**end if**

**end for**

*#Seleciona um percentual  $p$  de cada grupo identificado*

**for all**  $k \leftarrow (k = 0, \dots, k = n)$  **do**

$\lambda = \left(\frac{p}{size(K)}\right)$

$k' = k(\lambda)$

**end for**

**return** ( $k'$ )

---

A fim de garantir sua utilização em um contexto de fluxo de dados contínuo, o algoritmo DBSCAN foi adaptado com o fim de permitir a reunião de dados a partir de um sistema de *buffers*, formando uma janela temporal na qual os dados são reunidos e organizados durante um certo intervalo de tempo predefinido.

Conforme este intervalo é alcançado, os conjuntos de dados são processados e submetidos às outras camadas do *framework*, repetindo o processo enquanto existirem dados a serem processados. Os resultados obtidos pelo processamento do algoritmo são submetidos a um segundo nível de redução, nesse nível, uma amostra distribuída dos elementos de cada cluster identificado é selecionada para formar um subconjunto ainda menor que é então enviado pelas camadas adjacentes do *framework*.

A escolha por métodos de redução de dados baseados em agrupamento fundamenta-se na sua capacidade de identificar similaridades relacionadas, entre outras coisas, às características posicionais. Assim, as técnicas de agrupamento podem atuar na identificação de veículos próximos uns dos outros, interpretando estas características de proximidade como *clusters*. Uma descrição detalhada do funcionamento é apresentada no Algoritmo 5.

Analisando o Algoritmo 5 (DBSCAN), observa-se que os dados obtidos da janela temporal são submetidos a uma operação de agrupamento por similaridades para um certo intervalo de tempo ( $\Delta t$ ). O processo se inicia visitando um ponto aleatório  $P \in D$  que não foi visitado anteriormente. Todos  $P \in \epsilon.\textit{neighborhood}$  são recuperados e se  $\sum_{i=0}^{i=n} P > \textit{minPts}$ , um *cluster* é formado. Caso contrário,  $P$  é declarado como ruído.

Cabe observar, entretanto, que um ponto de ruído  $P$  pode ser incorporado como parte de outros *clusters* para um tamanho diferente de  $\epsilon.\textit{neighborhood}$  em futuras iterações. Este processamento tem por fim aferir a densidade dos grupos por meio das análises das similaridades posicionais e de velocidade. O processo continua até que todas as densidades de *clusters* sejam encontradas. Ao final, seleciona-se um percentual dos elementos que será incorporado ao conjunto de dados resultante. A Figura 4.4 ilustra como o processo de composição dos *clusters* corre em um modelo de fluxo de dados contínuos.



Figura 4.4: Mapeamento dos clusteres utilizando o algoritmo DBSCAN em fluxo de dados contínuo.

Analisando a Figura 4.4, observa-se que o algoritmo DBSCAN é capaz de identificar o congestionamento representado no simulador *Simulation of Urban Mobility* (SUMO). A identificação do congestionamento ocorre após o processo de redução de dados e a cada janela temporal, novos conjuntos são identificados e os cenários de tráfego são dinamicamente atualizados. Este processo ocorre de forma continuada enquanto existirem dados a serem processados.

#### 4.6.1.6 OPTICS

O algoritmo OPTICS (ANKERST et al., 1999) atua armazenando resultados baseados na ordem de processamento dos pontos, esta estratégia procura ordenar os pontos linearmente de forma que os pontos mais próximos tornam-se vizinhos de ordenação. Para cada ponto, têm-se uma distância armazenada a qual representará a densidade para um *cluster* atendendo ao critério de *MinPts*. Uma descrição detalhada é apresentada no Algoritmo 6.

---

#### Algorithm 6: OPTICS Adaptado

---

```

Require: Set of stream points ( $D$ ) containing all
positional vehicle information in the time
interval  $\Delta t$ , as well as  $MinPts$  and
NeighborRadius  $\varepsilon$ 
Initialize clusters  $k = Unprocessed$ 
for all points  $P \in D$  do
     $pt.reachable\_dist = undefined$ 
end for
for all unprocessed point  $P \in D$  do
     $Nbrs = getNbrs(P, \varepsilon)$ 
    mark  $P$  as processed
     $k'.append(P)$ 
    if  $core\_dist(P, \varepsilon, MinPts) \neq undefined$  then
         $Seeds =$  empty priority queue
         $update(Nbrs, P, Seeds, \varepsilon, MinPts)$ 
        for all  $q \in Seeds$  do
             $Nbrs' = getNbrs(q, \varepsilon)$ 
            mark  $q$  as processed
             $k'.append(q)$ 
            if  $core\_dist(q, \varepsilon, MinPts) \neq undefined$  then
                 $update(Nbrs', q, Seeds, \varepsilon, MinPts)$ 
            end if
        end for
    end if
end for
end for
return ( $k'$ )

```

---

Na presente pesquisa, o algoritmo OPTICS foi adaptado para atuar em um contexto de fluxo de dados contínuo. Assim, dados oriundos da camada de mobilidade são reunidos e organizados durante um certo intervalo de tempo e então submetidos ao processamento. Os resultados deste processamento são submetidos a um segundo nível de redução de dados no qual certos elementos de cada *cluster* são selecionados, formando um subconjunto representativo do conjunto original.

O primeiro passo do algoritmo 6 adaptado é inicializar a distância de todos os pontos em  $D$  com *undefined*. Então, para cada ponto ainda não processado, será verificado se todos os vizinhos de  $P$  são capturados dada uma distância definida por  $\varepsilon$ . Subsequentemente, verifica-se se o ponto é marcado como processado e na linha 7 ele é adicionado a uma lista ordenada  $k'$  (que é a saída do algoritmo). Logo após, verifica-se que o ponto  $pt$  não é um ruído, ou seja, se a distância do núcleo de  $P$  não é *undefined*. Se for verdade, uma fila de prioridade é inicializada e atribuída a *Seeds*. Na linha 10, é invocada a função *update*, responsável por atualizar a distância alcançável de cada vizinho de  $P$ , passando como parâmetros os vizinhos de  $P$  (*Nbrs*),  $P$  em si, a fila *Seeds* inicializada anteriormente,  $\varepsilon$  e *MinPts*. Após atualizar os pontos vizinhos e preencher a fila *Seeds*, cada item dessa fila é processado. Os vizinhos para cada  $q$  item em *Seeds* são recuperados, marcados como processados e adicionados a lista ordenada  $k'$ . Na linha 15, verifica-se se a distância do núcleo de  $q$  também não é ruído. Em caso afirmativo, a função de atualização é novamente invocada para processar  $q$  e seus vizinhos *Nbrs'*. No final, o algoritmo OPTICS retorna uma lista ordenada  $k'$  com todos os pontos processados  $P$  e  $q$ .

#### 4.6.1.7 X-Means

Embora o K-means seja amplamente usado para detectar agrupamentos, este algoritmo requer um certo conhecimento sobre os conjuntos de dados para definir um *k-number* antes do processo de agrupamento. Assim, optou-se nesta proposta por incorporar o algoritmo X-Means. O algoritmo é inicializado com 2-Means (K-Means com  $k = 2$ ) como uma sub-rotina que tenta subdividir repetidamente os agrupamentos, mantendo as melhores divisões resultantes até que o *Bayesian Information Criterion* (BIC) (SCHWARZ, 1978) seja alcançado. O BIC é baseado na *log* da probabilidade do conjunto de dados e é encontrado na Equação  $BIC(j) = \log M_j(D) - \frac{1}{2}kp_j \log n$  Onde  $M_j(D)$  é a probabilidade máxima de  $D$  usando o modelo  $j$ ;  $kp_j$  é o número de parâmetros do modelo e  $n = |D|$ . De forma similar aos demais algoritmos abordados nessa pesquisa, o algoritmo X-Means, baseado na métrica BIC, também foi adaptado para atuar em um contexto de fluxo de dados contínuo, sua estrutura é descrita no Algoritmo 7.

O algoritmo X-Means utiliza um processo de divisões simultâneas dos grupos, a fim de alcançar a melhor distribuição possível para um certo conjunto de dados. Assim, este processo ocorre com base no fluxo de dados distribuídos pela janela temporal. Nesse contexto, os dados de tráfego são agrupados em um intervalo de tempo ( $\Delta t$ ) usando 2-Means. Essa divisão inicial culminará na formação de dois agrupamentos. Em termos práticos, estes dois grupos representam os focos de maior densidade veicular observados na primeira iteração.

---

**Algorithm 7:** X-Means Adaptado: Redução de Dados em Fluxo Contínuo

---

**Require:** Conjunto de pontos obtidos no fluxo de dados ( $D$ ) contendo todas as informações posicionais do veículo no intervalo de tempo  $\Delta t$

**Require:**  $k$

**Require:** Valor em percentual  $P$  dos elementos selecionados em cada *cluster*

**Ensure:** Conjunto de agrupamentos ( $k'$ ) atribuídos para cada  $\Delta t$

```

while  $\Delta t$  do
   $buffer \leftarrow D$ 
end while
 $D \leftarrow buffer$ 
Inicializa o conjunto de clusters  $k = 2$ 
while  $\log M_j(D) - \frac{1}{2}kp_j \log n$  OR  $K > D$  do
  if  $BIC(j) == True$  then
     $K \leftarrow BIC(j)$ 
    for all novo ponto  $P \in K$  do
      if P is visited then
         $P_i = P_{i+1}$ 
      else
         $Update(P)$ 
         $P \leftarrow visited$ 
      end if
    end for
  else
     $Next(k)$ 
  end if
end while
for all  $k \leftarrow (k = 0, \dots, k = n)$  do
   $\lambda = (\frac{p}{size(K)})$ 
   $k' = k(\lambda)$ 
end for
return ( $k'$ )

```

---

Em seguida, 2-Means é novamente executado em cada um dos agrupamentos formados na primeira iteração. Após este processo, o algoritmo define dois centroides deste novo grupo com base no centroide do *cluster* original e compara com os elementos contidos no raio de cobertura. Se algum dos elementos do grupo inicial não tiver uma boa representação dos dados, de acordo com a métrica BIC, estes agrupamentos serão substituídos por seus “filhos”. Se nenhum *cluster* for pior do que seu filho, um membro do *cluster* é escolhido para representar os dados com base nas métricas adotadas. O processo é repetido até que o número de *clusters* exceda o número máximo fornecido como entrada.

## 4.6.1.8 HDBSCAN

O algoritmo HDBSCAN, proposto por (CAMPELLO; MOULAVI; SANDER, 2013), atua como uma extensão do algoritmo DBSCAN. Nesta abordagem, procura-se adicionar as características hierárquicas ao algoritmo original a partir da extração de grupos com base na sua estabilidade. O funcionamento do algoritmo HDBSCAN é ilustrado pelo Algoritmo 8

---

**Algorithm 8:** HDBSCAN Adaptado: Redução de Dados em Fluxo Contínuo
 

---

**Require:** Conjunto de pontos obtidos no fluxo de dados ( $D$ ) contendo todas as informações posicionais do veículo no intervalo de tempo  $\Delta t$

**Require:**  $k$

**Require:** Valor em percentual  $P$  dos elementos selecionados em cada *cluster*

**Ensure:** Conjunto de agrupamentos ( $k'$ ) atribuídos para cada  $\Delta t$

```

while  $\Delta t$  do
   $buffer \leftarrow D$ 
end while
 $D \leftarrow buffer$ 
while  $CoreDistance(D, Min_{pts})$  do
   $G \leftarrow MST(D)$ 
  for all  $G' \in G$  do
     $MST_{ext} \leftarrow MST(G')$ 
  end for
   $tree \leftarrow MST_{ext}$ 
  while  $tree.root$  do
     $tree.root \leftarrow tree.child()$ 
  end while
   $(HDB, noise) \leftarrow Next(tree, noise)$ 
   $k \leftarrow HDB$ 
end while
for all  $k \leftarrow (k = 0, \dots, k = n)$  do
   $\lambda = (\frac{p}{size(K)})$ 
   $k' = k(\lambda)$ 
end for
return ( $k'$ )

```

---

Conforme mencionado no Capítulo 2, a estrutura do HDBSCAN foi projetada para atuar mesmo em ambientes com densidades variadas, este processo é possível através do cálculo das distâncias para o núcleo ou *core distance*. Essa relação assegura a implementação de valores variáveis para épsilon, garantindo que o raio varie em função dos níveis de dispersão ou agrupamentos identificados. A formação hierárquica é construída com base na equação  $maxd_{core}(x_p), d_{core}(x_q), d(x_p, x_q)$ , onde  $d(x_p, x_q)$  se refere a medida

de distância escolhida, a exemplo da distância euclidiana, e a distância para o núcleo (*Core Distance*) é definida como a distância de um ponto para o seu vizinho mais próximo baseado no  $Min_{pts}$ , previamente configurado. Essa abordagem separa pontos esparsos uns dos outros a uma distância máxima que equivale a *core distance*. Essa característica do HDBSCAN tem por função tornar o agrupamento mais consistente, reduzindo a presença de ruídos.

Adicionalmente, toda a estrutura do algoritmo está submetida a uma adaptação com o fim de garantir a sua atuação em um contexto de fluxo de dados contínuo. De forma similar aos demais algoritmos, essa adaptação preconiza o uso de janelas temporais e reprocessamentos das saídas, com o fim de identificar um subconjunto de dados menor e igualmente representativo.

Analisando o Algoritmo 8, observa-se que o processo se inicia com o cálculo das distâncias do núcleo para todos os objetos de dados dentro de uma configuração mínima de pontos  $Min_{pts}$ . Em seguida, são calculados os valores para MST (*Mutual Reachability Graph*) para o conjunto de dados analisado. Para cada vértice será adicionada uma aresta a partir de uma extensão do  $MST_{ext}$ . Esse novo conjunto compõe uma visão hierárquica inicial do conjunto de dados original. Para cada raiz da árvore, serão atribuídos todos os objetos com rótulos similares. Em seguida, serão removidas as arestas do  $MST_{ext}$  em ordem decrescente de pesos. Antes de cada remoção, é definido o valor para cada nível hierárquico com base em seus pesos. Após cada remoção, são atribuídas as relações conectando-as a outros componentes da árvore para obter o próximo nível hierárquico. Por fim, um novo rótulo será atribuído caso tenha pelo menos uma aresta. Elementos que não forem alocados serão interpretados com ruído.

#### 4.6.2 Level of Service - LOS

Após os algoritmos de redução de dados de tráfego serem empregados pelo módulo de monitoramento, os dados de tráfego reduzidos são encaminhados para um novo processo, a classificação. Conforme já mencionado, um dos principais objetivos deste trabalho é minimizar os custos de comunicação aliviando os *links* LTE-5G por meio da redução de dados na névoa, no entanto, após a redução, estes conjuntos de dados resultantes devem manter-se igualmente representativos, a métrica *Level of Service* (LOS) foi utilizada para mensurar este grau de representatividade. A Tabela 4.1 ilustra os intervalos por classificação.

Tabela 4.1: Classificação LOS/HCM

$(w_i)$	LOS	Classificação do Congestionamento de Tráfego
[0, 0.15]	A	Free-flow
[0.15, 0.33]	B	Reasonably free-flow
[0.33, 0.50]	C	Stable-flow
[0.50, 0.60]	D	Approaching unstable-flow
[0.60, 0.70]	E	Unstable-flow
[0.70, 1.00]	F	Breakdown-flow

A classificação por nível de serviço (LOS) foi criada pelo *United States Transportation Research Board* (BOARD, 2010) na forma de equações matemáticas com o fim de estimar o nível de serviço do tráfego. Para isso, a abordagem consiste na definição de seis diferentes classes que fornecem uma referência para medições usadas para descrever as condições do fluxo de tráfego em um certo cenário.

Um algoritmo para cálculo do nível de serviço foi implementado como forma de realizar a classificação dos dados recebidos na nuvem, por meio do fluxo de dados contínuo enviado pela névoa. O algoritmo de classificação por nível de serviço é iniciado com a criação de um recorte nos dados sobre o qual os valores de densidade serão aferidos. Neste contexto, os dados destes subconjuntos são ordenados de forma que o processo de classificação avalie as mudanças de densidade ocorridas durante a execução dos cálculos sobre estes dados. Após a criação do *buffer*, os dados de cada ponto são processados com base na equação  $w_i = 1 - \frac{s_i^{media}}{s_i^{max} \times d_i} \mid d_i > 0$  e para cada ponto, um peso  $w_i$  é atribuído classificando o elemento com base no seu nível de serviço.

Por fim, uma lista contendo os elementos classificados é fornecida como resultado desta operação. Esta lista, em última análise, representa a capacidade de cada um dos algoritmos incorporados à camada de monitoramento de identificar o comportamento do tráfego nas rodovias monitoradas. Estes conjuntos de dados, classificados em fluxo contínuo, são utilizados como base para as estratégias de roteamento. O módulo de roteamento é apresentado na seção seguinte.

### 4.6.3 Módulo de Roteamento

Conforme os dados de classificação LOS são enviados da névoa para a nuvem, novos panoramas são formados acerca do estado do tráfego nas vias monitoradas. Assim, decisões tomadas na camada da nuvem podem requerer eventuais intervenções no fluxo de tráfego. As requisições emitidas pela camada de nuvem são então retornadas à camada da névoa por meio de uma interface comum de integração. Analogamente, as requisições recebidas na camada da névoa são processadas no módulo de roteamento que tem a função de propagar as decisões tomadas na nuvem para o cenário urbano. Todo o processo ocorre a partir da execução de um algoritmo estruturado para realizar uma abordagem baseada em *k-shortest path*. Nesse contexto, os dados submetidos ao algoritmo são analisados e um conjunto de rotas priorizadas é gerado para cada veículo solicitante em um intervalo de tempo  $T_n$ .

O processo de roteamento é inserido no fluxo de atividade do framework da seguinte forma: os dados enviados pelo ambiente de mobilidade são coletados e submetidos aos processos de redução de dados, mencionados na seção anterior. Após esta fase, os dados reduzidos são enviados para a nuvem onde são reunidos e analisados, configurando uma visão geral do cenário urbano. Com base nessas análises, os alertas de densidade são emitidos para a camada de névoa, na qual novas rotas são geradas e enviadas aos veículos na camada de mobilidade.

Entretanto, cabe citar que o cenário urbano está em constante atualização, uma vez que as densidades das rodovias variam com frequência e esta variação modifica a distribuição das rotas em cada mudança de cenário. Para superar este desafio, o algoritmo

*k-shortest path* foi integrado ao *framework* utilizando uma abordagem baseada em fluxo de dados contínuo. Assim, conforme novos dados são recebidos, novas análises são realizadas, permitindo a emissão de novas sugestões de rotas, de forma dinâmica e contínua, desde a origem até o destino, para cada veículo presente no cenário. O Algoritmo 9 ilustra este mecanismo de distribuição de rotas.

---

**Algorithm 9:** K-Shortest Path: Adaptado de (YEN, 1971)

---

**Require:** *graph* caminhos da rede,  
**Require:** *source* ponto de origem,  
**Require:** *target* posição de destino,  
**Require:** *k* número de caminhos esperados,  
**Require:** *losWeight* peso das vias,  
 $G[0] \leftarrow Dijkstra(Graph, source, sink)$   
 $TMP \leftarrow []$   
**for**  $k \leftarrow 1$  to  $k$  **do**  
  **for**  $i \leftarrow size(G[k - 1]) - 2$  **do**  
     $spurNode \leftarrow G[k - 1].node(i)$   
     $rootPath \leftarrow G[k - 1].nodes(0, i)$   
    **for**  $pathp \in G$  **do**  
       $removegraph.edge()$   
    **end for**  
     $spurPath \leftarrow Dijkstra(Graph, spurNode, sink)$   
     $pathbyWheight \leftarrow VerifyWeight(spurPath, losWeight)$   
     $TMP.append(pathbyWheight)$   
  **end for**  
   $L[k] \leftarrow TMP[0]$   
**end for**  
**return**  $L$

---

Analisando o Algoritmo 9, pode-se observar que todo processo se inicia com a identificação do caminho mais curto entre uma origem e um destino em uma rede de tráfego descrita por um grafo direcional  $G(u, v)$ . A partir deste ponto, todas as possíveis rotas entre estes dois pontos são analisadas, a fim de identificar o caminho mais curto subsequente.

Conforme os caminhos mais curtos são identificados, as rotas são submetidas a uma verificação dos pesos das rodovias que compõem as rotas selecionadas. Estes pesos são fornecidos ao algoritmo e representam a métrica LOS encontrada em cada trecho. A métrica LOS fornece um valor entre 0 e 1 que indica o nível de congestionamento de cada trecho das rodovias monitoradas.

À medida que os resultados são processados, as rotas são armazenadas de forma ordenada em uma estrutura de dados adicional. Ao final, com todas as possíveis  $k$  rotas alcançadas, os dados são consolidados em uma lista que é então enviada como retorno do algoritmo.

A eficiência deste modelo traz benefícios relacionados não só à dinâmica do fluxo de tráfego, mas também em áreas tangenciais, conhecidamente impactadas pela redução do tráfego, como a possível redução no consumo de combustível, reduções nos níveis de  $CO_2$  e melhora das condições de vida dos cidadãos, sendo este último, uma das principais razões para a construção de sistemas inteligentes.

A camada da nuvem, abordada na próxima seção, é responsável por unificar os conjuntos de dados provenientes das diferentes RSUs, essa estratégia tem por fim formar uma visão geral do estado do tráfego de modo a auxiliar no processo e tomada de decisões não só relacionadas à emissão dos alertas de densidade, mas também sobre quais veículos requerem ou não a emissão de novas rotas. O peso das rotas cumpre um indicador importante nesse processo decisório.

## 4.7 CAMADA DA NUVEM

A camada de computação em nuvem cumpre um papel essencial na construção do sistema de transporte inteligente, proposto neste trabalho. Sua função principal é descrever, a partir dos diferentes fluxos dados recebidos da camada de névoa, um panorama geral acerca do estado do tráfego nas rodovias monitoradas e sobre este mesmo panorama, definir estratégias que possam conduzir à melhoria do fluxo de tráfego.

Neste contexto, a camada de nuvem atua em três frentes distintas, porém, complementares, a saber: 1) Unificação do fluxo de dados recebidos pelos diferentes nós dispostos na borda da rede; 2) Construção de um modelo unificado contendo os diferentes níveis de fluxo de tráfego obtidos da camada de névoa; 3) Tomada de decisões baseadas nos cenários observados. Conforme mencionado, estas decisões podem ser caracterizadas por alertas acerca de rodovias em processo de formação de congestionamento, bloqueio de rodovias ou mesmo sugestões de roteamento para os veículos dispostos no espaço urbano. No contexto deste trabalho, a definição de congestionamento de tráfego é a seguinte:

**Definição 14.** *Seja um segmento de estrada  $e_i$  modelado como um conjunto de pesos  $W = \{(w_1, t_1), (w_2, t_2), \dots, (w_{|W|}, t_{|W|})\}$  representando a condição de tráfego  $w_j$  ao longo do tempo  $t_i$ , em que  $w : E \rightarrow \mathbb{R}_+^*$ , onde, cada  $w_j$  está tratando da condição de tráfego no momento  $t_i$  para um certo segmento de estrada  $e_i$*

A camada de computação em nuvem recebe um fluxo de dados da camada anterior por meio de uma *interface* comum que as integra. Esta *interface* utiliza o protocolo de comunicação IEEE 802.11p e, assim como as *interfaces* anteriormente apresentadas, também se utiliza de um padrão de comunicação para troca de mensagens. Este padrão é reconhecido por todas as camadas do modelo.

Conforme os conjuntos de dados são recebidos a partir dos diferentes nós da névoa, um processo de integração destes dados é acionado. Este processo consiste em unificar, em uma estrutura de dados comum, os conjuntos oriundos dos diversos nós da camada de névoa. O conjunto de dados resultante desta operação pode ser descrito como uma visão integrada do fluxo de tráfego em constante atualização e sobre o qual serão realizados os procedimentos para a tomada de decisão.

Os dados unificados na camada de computação em nuvem são submetidos ao processo de agrupamento e análise de dados. Neste processo, os dados são analisados de uma perspectiva geográfica e espacial com a finalidade de obter uma visão na qual seja possível verificar geograficamente os pontos de retenção e possíveis ações de contenção para mitigar ou eliminar o congestionamento de tráfego. Este processo apresenta subsídios para a construção de um ambiente de visualização que fornece, em tempo próximo ao real, o estado do tráfego nas rodovias monitoradas.

Por fim, têm-se os processos responsáveis pela emissão dos alertas de densidade e sugestões de roteamento. Com base no cenário observado em um certo instante de tempo, pode ser necessário realizar certas intervenções como forma de mitigar a formação e congestionamento e garantir níveis de fluxo de tráfego aceitáveis de acordo com o padrão de classificação LOS. Nesse contexto, conforme os dados do ambiente de mobilidade são processados pelos algoritmos de redução de dados, estes resultados são encaminhados para cálculo dos níveis de serviço e estas análises deverão indicar os pontos de retenção que serão utilizados nas sugestões de novas rotas. A Figura 4.5 ilustra o processo de classificação por nível de serviço em fluxo contínuo e os veículos que receberam novas sugestões de rotas com base nos pontos de retenção.

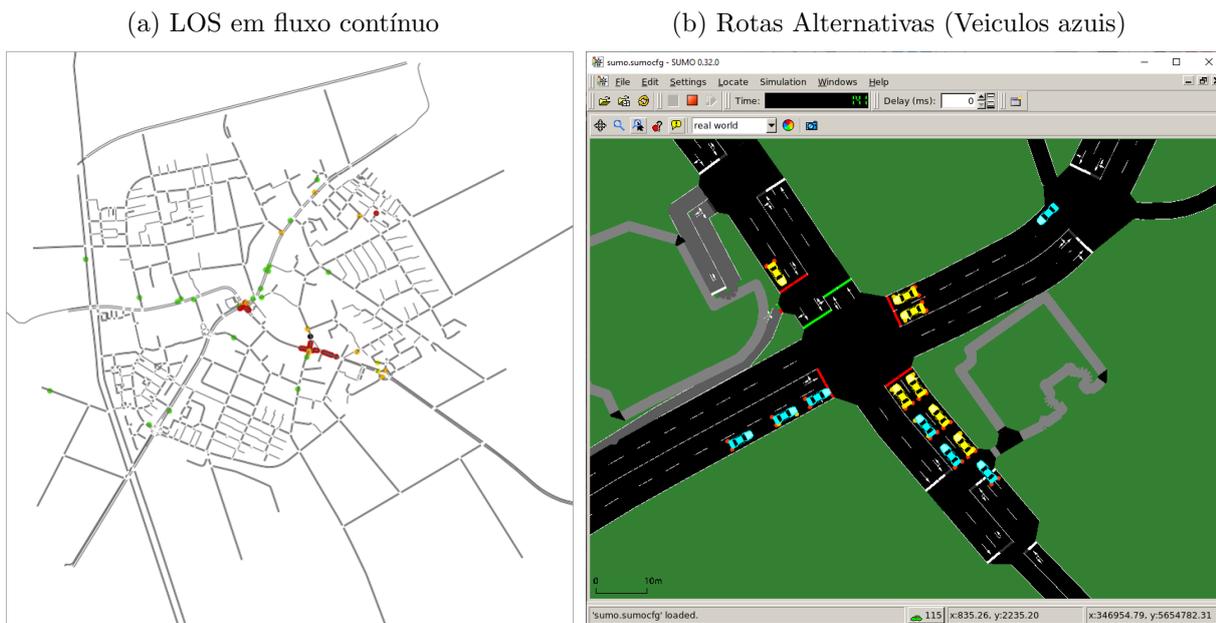


Figura 4.5: Classificação LOS como base para a distribuição de rotas alternativas

Os níveis de intervenções realizadas no cenário podem variar em função do comprometimento do fluxo de tráfego. O sistema de monitoramento baseado na métrica LOS garante a identificação de pontos de retenção tão logo eles se formem em certos segmentos da rodovia. Nestes casos, alertas são emitidos para todos os veículos que desejam utilizar a referida via, informando a melhor rota alternativa. Na Figura 4.5b, durante a simulação, veículos que recebem essa sugestão de rota são marcados em azul. A emissão de alertas indica a redução dos níveis de fluxo e poderá motivar os motoristas a evitar

trechos de maior congestionamento. Analogamente, a paralisação inesperada do trânsito pode requerer outros níveis de intervenção. Nestes casos, alertas de bloqueio podem ser emitidos como forma de garantir não só a fluidez do tráfego, mas principalmente o impacto desta ação em outros segmentos da rodovia no cenário urbano.

As ações do serviço de alertas e roteamento podem ser realizadas de forma autônoma, na medida em que um *threshold* pode ser definido como forma de acionar certas intervenções ou ainda por especialista. Neste caso, a presente proposta utilizou-se da classificação por nível de serviço para definir os pesos às rodovias e calcular um *threshold* dinâmico capaz de acompanhar a dinâmica do tráfego nas rodovias monitoradas.

## 4.8 CONSIDERAÇÕES DO FINAIS

Este capítulo apresentou o *framework* DaRkaM, sua arquitetura multicamadas, componentes e estratégias para processamento de redução de dados na névoa. Todo conceito proposto pelo *framework* considera um ambiente onde há um fluxo de dados contínuo, no qual os dados são transferidos e processados em cada uma das camadas do modelo, fornecendo os meios para realizar três objetivos fundamentais: a) detecção de congestionamento de tráfego, b) redução de dados e c) intervenção no sistema de tráfego por meio do re-roteamento de veículos.

Essas operações são estruturadas considerando uma arquitetura composta de três camadas, nesse contexto, têm-se a 1) camada de mobilidade, na qual uma soma de simuladores opera em conjunto para fornecer um sistema de tráfego, redes e infraestrutura de rodovias; 2) camada da névoa, responsável, entre outras coisas, por hospedar os algoritmos de redução de dados utilizados nos experimentos; 3) camada e nuvem, responsável por analisar os dados, gerando alertas que deverão fornecer rotas aos veículos na camada de mobilidade. Adicionalmente, a fim de estabelecer uma análise comparativa entre as diferentes estratégias de redução, um conjunto de abordagens foi implementado e/ou adaptado para compor o núcleo de redução de dados do *framework*, proposto nessa pesquisa. Nesse contexto, foram analisados os algoritmos baseados em amostragem simples como, 1TO2, Random e LImite, além dos algoritmos baseados em agrupamento: X-Means, DBSCAN, OPTICS e HDBSCAN.

No próximo capítulo, serão apresentados os resultados obtidos a partir das análises dos algoritmos selecionados frente ao algoritmo *Baseline*. Essa métrica fornecerá uma base sólida para estabelecer comparações entre as estratégias de identificação de congestionamentos e gerenciamento do tráfego, uma abordagem amplamente utilizada para analisar as variações frente às diferentes abordagens implementadas nessa pesquisa (SOUZA; VILLAS, 2016; SOUZA et al., 2016).



## METODOLOGIA DOS EXPERIMENTOS E ANÁLISE DOS RESULTADOS

Este capítulo tem por objetivo analisar, por meio de uma avaliação de desempenho, a eficiência do *framework* **DaRkaM** no que se refere a sua atuação no monitoramento, detecção e re-roteamento em um cenário de congestionamento de tráfego, avaliando, entre outros aspectos, as diferentes estratégias para seleção de dados e o grau de representatividade dessas estratégias na tomada de decisão.

As observações e discussões abordadas nesta seção passam pela avaliação de oito algoritmos, dos quais quatro são baseados em agrupamento, três utilizam uma abordagem baseada em seleção de amostragem e um algoritmo *Baseline* utilizado nas medições de acurácia. Entre os algoritmos de agrupamento estão o **X-Means** (PELLEG; MOORE, 2000), **DBSCAN** (ESTER et al., 1996), **OPTICS** (ANKERST et al., 1999) e **HDSCAN** (CAMPELLO; MOULAVI; SANDER, 2013), onde procurou-se analisar seus desempenhos, não só relacionados a redução de dados, mas também a capacidade destes algoritmos de estabelecerem níveis de acurácia representativos, frente a um conjunto de dados original. As estratégias baseadas em agrupamento foram comparadas com outras abordagens descritas na literatura como **1to2**, **Rand** (TANIZAKI; WOLFSON, 2007) e **Lim** (WAHID; RAO; GOEL, 2019) como forma de posicionar esta proposta frente a outras abordagens discutidas pela comunidade científica.

As medições de acurácia foram realizadas utilizando uma base comparativa obtida por meio de um algoritmo descrito como *Baseline* (SOUZA; VILLAS, 2016), (SOUZA et al., 2016) e (SOUZA et al., 2018). Esta abordagem consiste na implementação de um algoritmo adicional que não se beneficia de qualquer estratégia de redução de dados, utilizando os dados brutos obtidos a partir da camada de mobilidade. Este algoritmo fornece, portanto, a linha de base que será utilizada para mensurar os níveis de precisão dos algoritmos no processo de identificação do grau de congestionamento de tráfego.

### 5.1 CONFIGURAÇÕES DO CENÁRIO E SIMULAÇÕES

Os simuladores utilizados foram o *framework* *Vehicle in Network Simulation* (VEINS) (VEINS, 2019), um framework de código aberto que implementa o protocolo padrão IEEE

802.11p para comunicação entre veículos, além de fornecer um modelo de obstáculo para atenuação de sinal. Para reproduzir os aspectos de mobilidade foi utilizado o simulador de mobilidade urbana SUMO (SUMO, 2019) que cumpre a função de simular o tráfego. Essa é uma solução de código aberto que forneceu os meios para modelar e manipular objetos no cenário rodoviário. Utilizou-se ainda o **OMNET! (OMNET!)** (OMNET++, 2019), um simulador de rede capaz de oferecer os meios para garantir a conectividade entre os elementos deste cenário urbano.

Cabe lembrar que o experimento foi estruturado em um ambiente de simulação para redes veiculares. Nesse contexto, os veículos emitem *beacons* que são capturados por uma *Roadside Unit* (RSU) mais próxima e são encaminhados para processamento na névoa. Analogamente, na camada de névoa, os beacons são processados e sobre eles são implementadas as estratégias de redução de dados propostas nessa pesquisa.

Durante os experimentos, assumiu-se a existência de um link de rede entre a névoa e a nuvem com uma taxa máxima de 340 (KB/s). Neste contexto, vale ressaltar que no cenário proposto para os experimentos foi utilizada uma *Roadside Unit* a serviço de todo o mapa, mas essa arquitetura pode ser modificada se operarmos com diferentes instâncias do *framework* para cada RSU.

O aumento no número de RSU implica no aumento da abrangência do tráfego observado nas simulações. Todos os principais parâmetros de simulação estão resumidos na Tabela 5.1.

Tabela 5.1: Parâmetros de Simulação

Parâmetros	Valores
Simulation Area	52.3 km <sup>2</sup>
Scenario	TAPASCologne
Boundary Box	(6.80580,50.99983) (6.84675,51.060)
Number of Vehicles	{100, 200 and 300}
Vehicles Speed	Mean: 30.05 km/h
Beacon Size	100 bytes
MAC Layer	IEEE 802.11p PHY
Mobility Simulator	SUMO 0.32.0
Vehicular Network Simulation	Veins 4.7
Discrete Event Simulator	OMNeT++ 5.3
Transmission Power	20 mW
Bit Rate	6 Mbps
Beacon Transmission Rate	1Hz

O ambiente de simulação estruturado para os experimentos utilizou como base um subconjunto do projeto TAPAS-Cologne <sup>1</sup> (UPPOOR et al., 2014). TAPAS-Cologne consiste em um projeto de código aberto que fornece um grande conjunto de dados em

<sup>1</sup><https://sumo.dlr.de/wiki/Data/Scenarios/TAPASCologne>

escala com o mais alto realismo para simulação veicular urbana baseada no simulador SUMO. O TAPAS Cologne utiliza um mapa realista da cidade de Colônia, Alemanha, obtido a partir do *OpenStreetMap*<sup>2</sup>, conforme pode ser observado na Figura 5.1.

Os veículos usados na simulação compartilham as mesmas características, como tamanho, média de velocidade e desvio padrão. Para realizar as simulações, foi selecionada uma região mais independente ou desconectada da região central do mapa, essa escolha é ilustrada na Figura 5.1. A escolha pelo recorte da região deu-se em função da área apresentar maior densidade veicular, vias longas e gargalos que poderiam contribuir para a geração de cenários mais realistas.

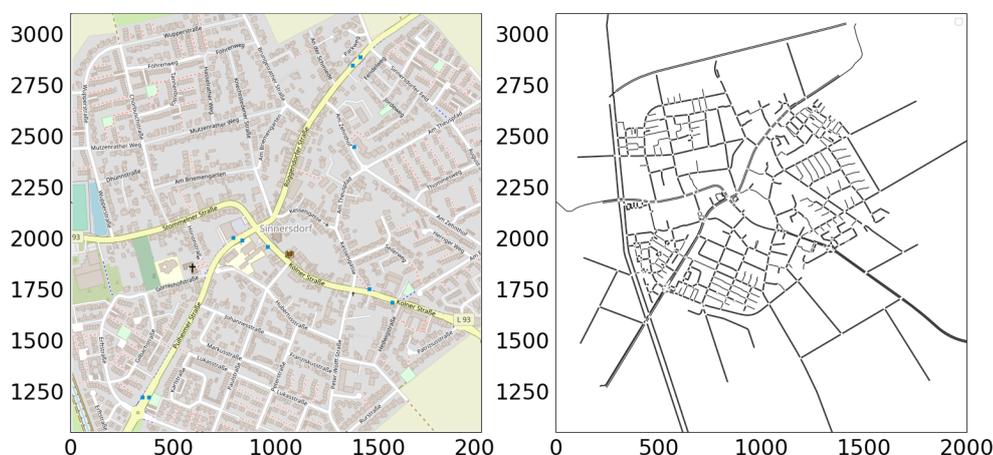


Figura 5.1: Recorte da Malha Viária de Cologne, Alemanha. Cenário do Ambiente de Simulação

Os experimentos realizados têm como fim observar o comportamento de oito diferentes algoritmos que apoiam o processo de detecção de congestionamento de tráfego, os quais foram explicados na seção 4.6. Para viabilizar esta análise, foram definidas rotas aleatórias para cada veículo. Além disso, diferentes tipos de densidades de veículos foram observadas. Estas densidades variaram entre 100, 200 e 300, as quais representam os veículos ativos no mapa durante o processo de simulação. As densidades de veículos foram escolhidas com o fim de caracterizar a ocorrência de congestionamento de tráfego no recorte da área referente a topologia rodoviária de Cologne.

A partir destes experimentos, foi possível analisar os níveis de acurácia na detecção de congestionamento de tráfego para cada uma das técnicas, separadamente. Neste cenário, cabe ressaltar que foi realizado um conjunto de experimentos com 10 replicações para cada algoritmo, utilizando um intervalo de confiança de 95%.

Os níveis de acurácia foram obtidos por meio da classificação do nível de serviço (LOS) e pelos impactos das estratégias de redução de dados no consumo de recurso de rede para

<sup>2</sup><https://www.openstreetmap.org>

cada uma das técnicas analisadas. Ainda sobre os experimentos, cabe citar que todas as abordagens aplicadas foram submetidas às mesmas condições arquiteturais, utilizando a mesma estrutura em camadas para gerenciamento do fluxo de dados contínuo e tomada de decisão em termos de re-roteamento e intervenções no cenário urbano.

## 5.2 VARIAÇÕES DE VELOCIDADE E TENDÊNCIAS

Esta seção tem por objetivo analisar o comportamento da velocidade média em km/h ao longo do tempo de simulação. Em função das variações nos tempos de simulação entre os experimentos, uma medida de normalização foi aplicada com o fim de garantir a fidelidade das análises e comparações a respeito das tendências observadas a partir do comportamento dos veículos. A normalização foi aplicada em todas as estratégias de redução analisadas: **Baseline**, **1to2**, **Rand**, **Lim**, **DBSCAN**, **X-MEANS**, **OPTICS** e **HDBSCAN**. Em todos os experimentos realizados, as densidades foram variadas entre 100, 200 e 300 veículos dispostos no cenário de mobilidade.

Com o fim de analisar o comportamento dos veículos nas rodovias, no que se refere as variações nas velocidades aferidas, foram definidos segmentos de rodovias específicas nas quais o monitoramento deste atributo foi coletado. Estas rodovias são descritas neste trabalho como zonas de interesse. Conforme pode ser observado na Figura 5.2, cada zona de interesse recebeu um número de referência (1, 2 e 3) que representa os respectivos segmentos. Esta numeração será utilizada durante as análises subsequentes.

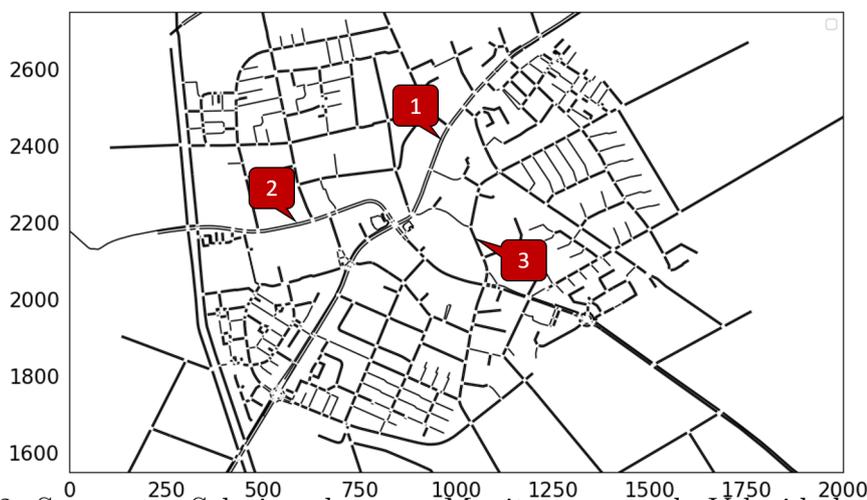


Figura 5.2: Segmentos Selecionados para Monitoramento da Velocidade no Tráfego

Os segmentos analisados neste experimento foram definidos com base em diferentes características. A motivação para tal escolha fundamenta-se na necessidade de identificar quais, entre os segmentos possíveis, poderiam fornecer melhores resultados comparativos em termos do monitoramento, detecção de congestionamento de tráfego e re-roteamento. Assim, a escolha do **segmento 1** deu-se em função deste ser um segmento mais longo, com maior número de intersecções e por tratar-se de uma via de acesso expresso a outros pontos do mapa. Adicionalmente, este segmento tem potencial para operar como uma via alternativa ao tráfego de outras regiões; O **segmento 2** representa uma artéria urbana

principal, trata-se de uma via extensa que promove a confluência do fluxo de tráfego para um ponto central no mapa. Este comportamento pode causar retenção neste segmento, propagando a eventual intensificação do tráfego nas rodovias secundárias. O **segmento 3** foi escolhido devido a sua intersecção como uma rota de trânsito rápido, é bidirecional e possui um gargalo com propensão a retenções de tráfego.

Embora este estudo acerca das variações e tendências relacionadas à velocidade abarque apenas as três zonas de interesse supracitadas, as classificações relativas ao nível de serviço LOS, discutidas na seção seguinte, evidenciarão uma análise mais ampla, envolvendo toda a extensão do mapa.

A fim de estabelecer um panorama geral acerca da análise e das velocidades dos veículos ao longo do processo de simulação, as Figuras 5.3, 5.4 e 5.5 ilustram este comportamento nos três segmentos de rodovias predefinidos. Além disso, os experimentos foram replicados variando a densidade dos veículos entre 100, 200 e 300, conforme descrito anteriormente.

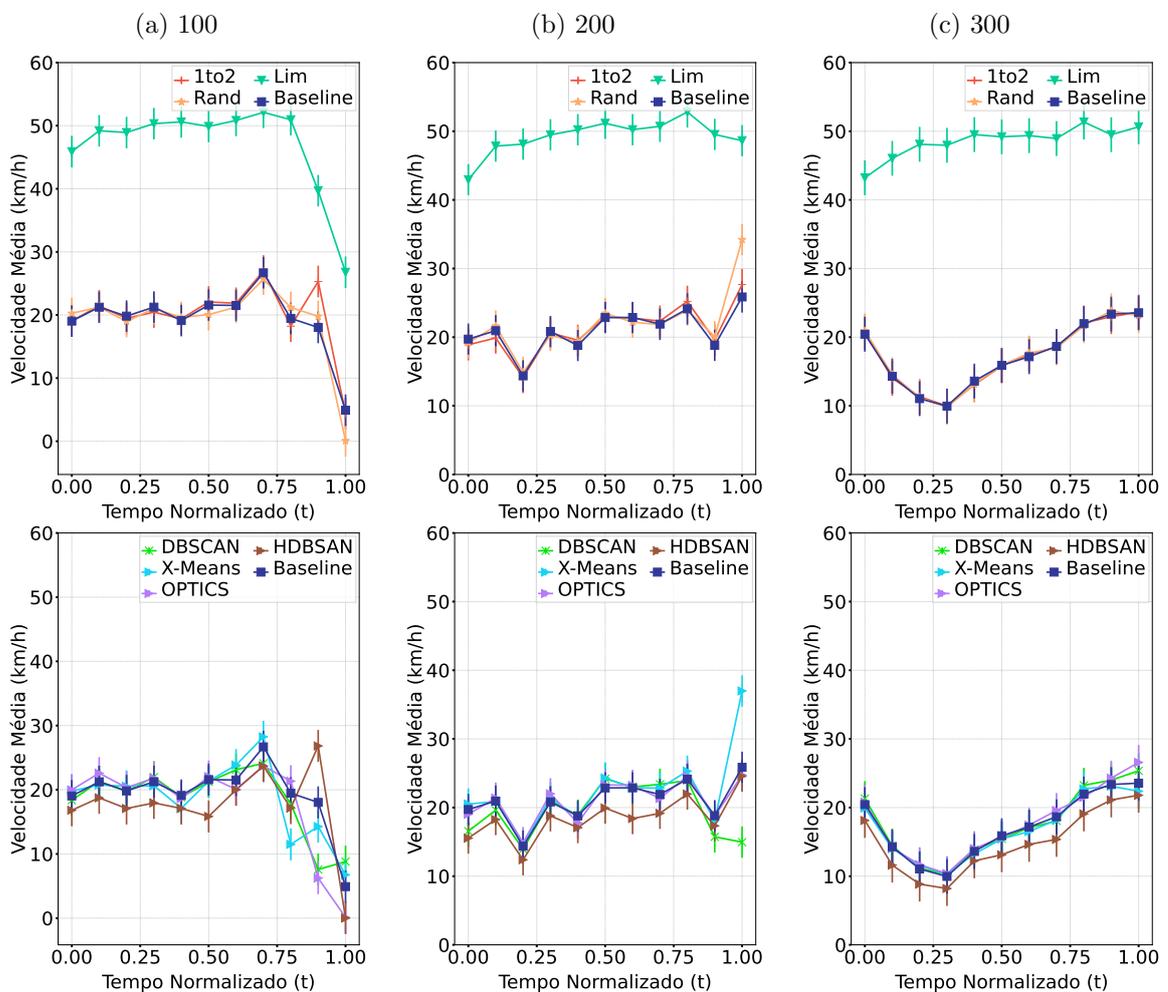


Figura 5.3: Análise da Velocidade ( $km/h$ ) no Segmento (1)

A Figura 5.3 descreve um comportamento semelhante para os algoritmos 1to2, Rand e *Baseline*, mostrando que as linhas de velocidade estão praticamente sobrepostas. Pode-se

observar ainda, que quando há uma variação sensível na tendência da velocidade aferida (Figura 5.3 a e c), há também uma sensível variação na tendência entre as técnicas de agrupamento e o algoritmo *Baseline*. Entretanto, cabe destacar que nestes casos o algoritmo HDBSCAN demonstra um ligeiro deslocamento quando comparado ao *Baseline*. Isso pode ser observado nas Figuras 5.3a e 5.3c. Este comportamento pode ser explicado pelo curto período de tempo em que a estrada se encontrava congestionada. Os picos de congestionamento de tráfego de curto período podem não ser detectados com alta precisão pelos algoritmos de agrupamento, uma vez que esse intervalo pode não ser suficiente para ter todos os dados de tráfego coletados para caracterizar a densidade espacial de um veículo.

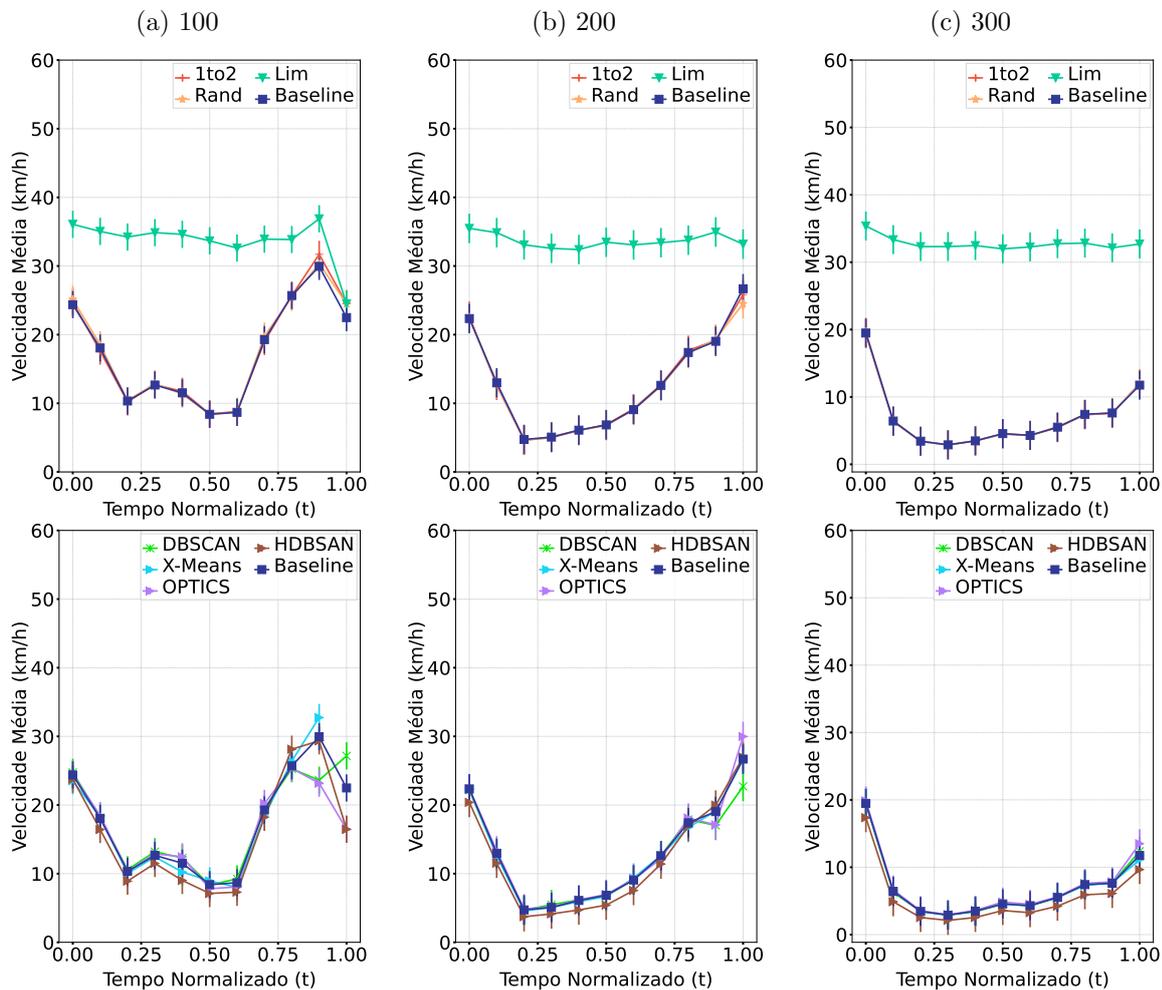


Figura 5.4: Análise da Velocidade ( $km/h$ ) no Segmento (2)

Cabe destacar ainda que para todos os cenários analisados, o algoritmo **Lim** apresenta menor acurácia frente ao *Baseline*. Esta diferença na acurácia pode estar relacionada ao limite estabelecido para a transmissão de dados de tráfego. Nesta abordagem, ambientes com alta variação de velocidade podem causar certa redução nos níveis de acurácia. Uma forma de mitigar este problema pode estar na utilização de um limite obtido, a partir

da média de dados de tráfego representados por um conjunto de dados histórico deste mesmo local. Com exceção de Lim, todos os outros algoritmos ilustrados na Figura 5.4 também mantiveram justo alinhamento no que se refere ao comportamento do algoritmo *Baseline*.

Analisando a Figura 5.5, especialmente nos experimentos que representam os níveis de densidades de veículos entre 200 e 300, pode-se observar que apesar da grande variação na velocidade, o DBSCAN é capaz de seguir a tendência do algoritmo *Baseline*. Pode-se observar ainda, que este processo ocorre mesmo em variações mais baixas, entre 10 e 20 km/h. Já o algoritmo **X-Means** e OPTICS apresentam uma leve perda de acurácia para 100 veículos, este efeito está relacionado, essencialmente, a dois fatores, a saber 1) baixo número de *clusters* formados durante o processo de simulação; 2) baixa representatividade da amostra em função do baixo número de elementos selecionados.

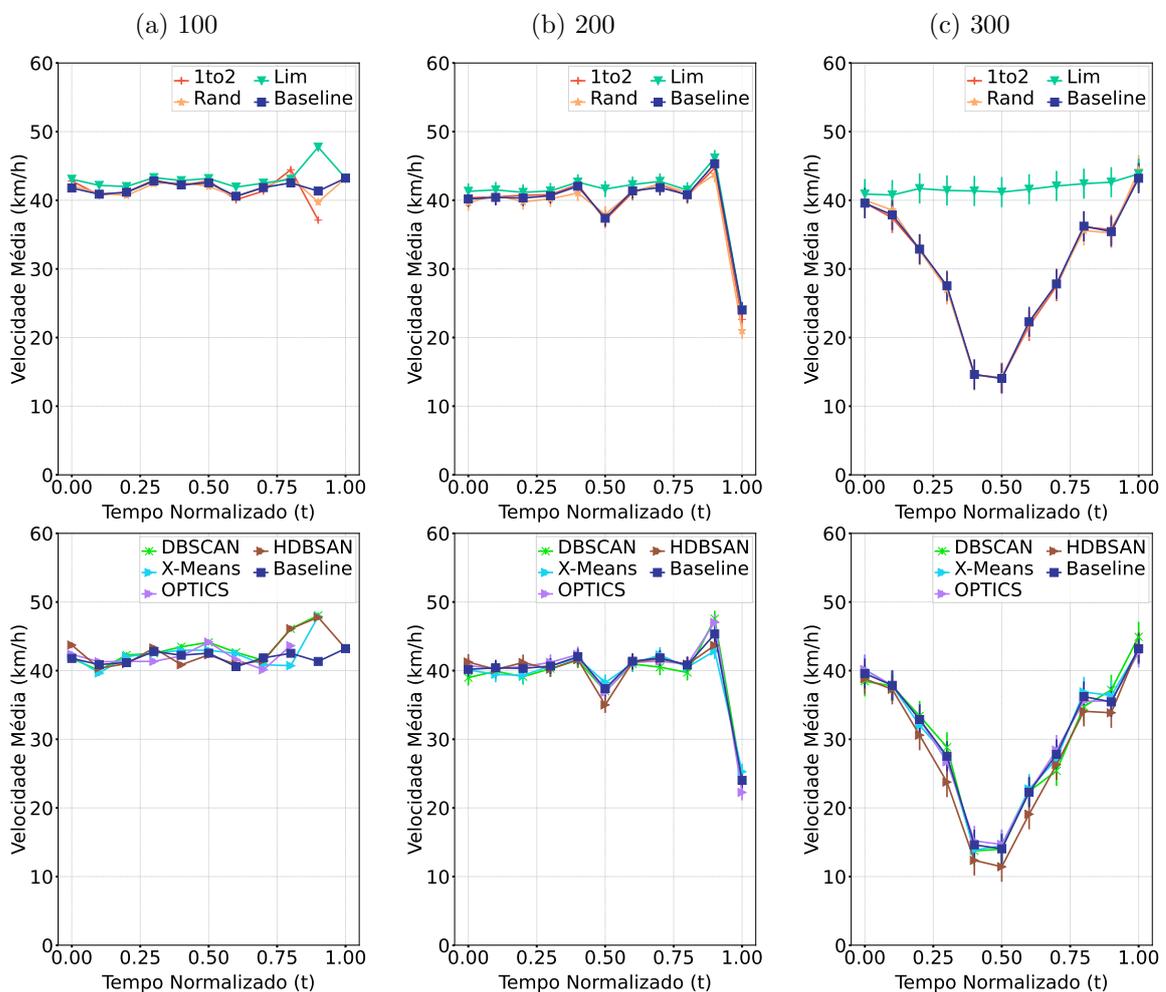


Figura 5.5: Análise da Velocidade ( $km/h$ ) no Segmento (3)

Os algoritmos DBSCAN e X-Means atuam destacando um percentual de elementos obtidos a partir de cada *cluster* formado para proceder com os mecanismos de redução de dados. Esta seleção, apesar de fornecer maiores chances de redução, em certas cir-

cunstâncias, pode não contemplar uma amostra representativa o suficiente para garantir o acompanhamento contínuo do fluxo de dados ao longo dos experimentos. Este cenário torna-se mais crítico quando o volume de dados analisados é reduzido. Este desafio pode ser mitigado com a definição de técnicas de seleção capazes de fornecer um conjunto resultante mais representativo frente ao conjunto de dados original.

A análise de velocidade é essencial para demonstrar o comportamento dos dados de tráfego que cada algoritmo da camada da névoa fornece para a nuvem. Com este tipo de informação de tráfego, o *framework* DaRkaM pode ser capaz de detectar a fonte e classificar o nível de congestionamento de acordo com sua severidade. Esta classificação passa pela definição do Nível de Serviço (LOS). A seguir, as Figuras 5.6, 5.7 e 5.8 ilustram os resultados relativos à aplicação do nível de serviço para os mesmos segmentos destacados nesta etapa do experimento.

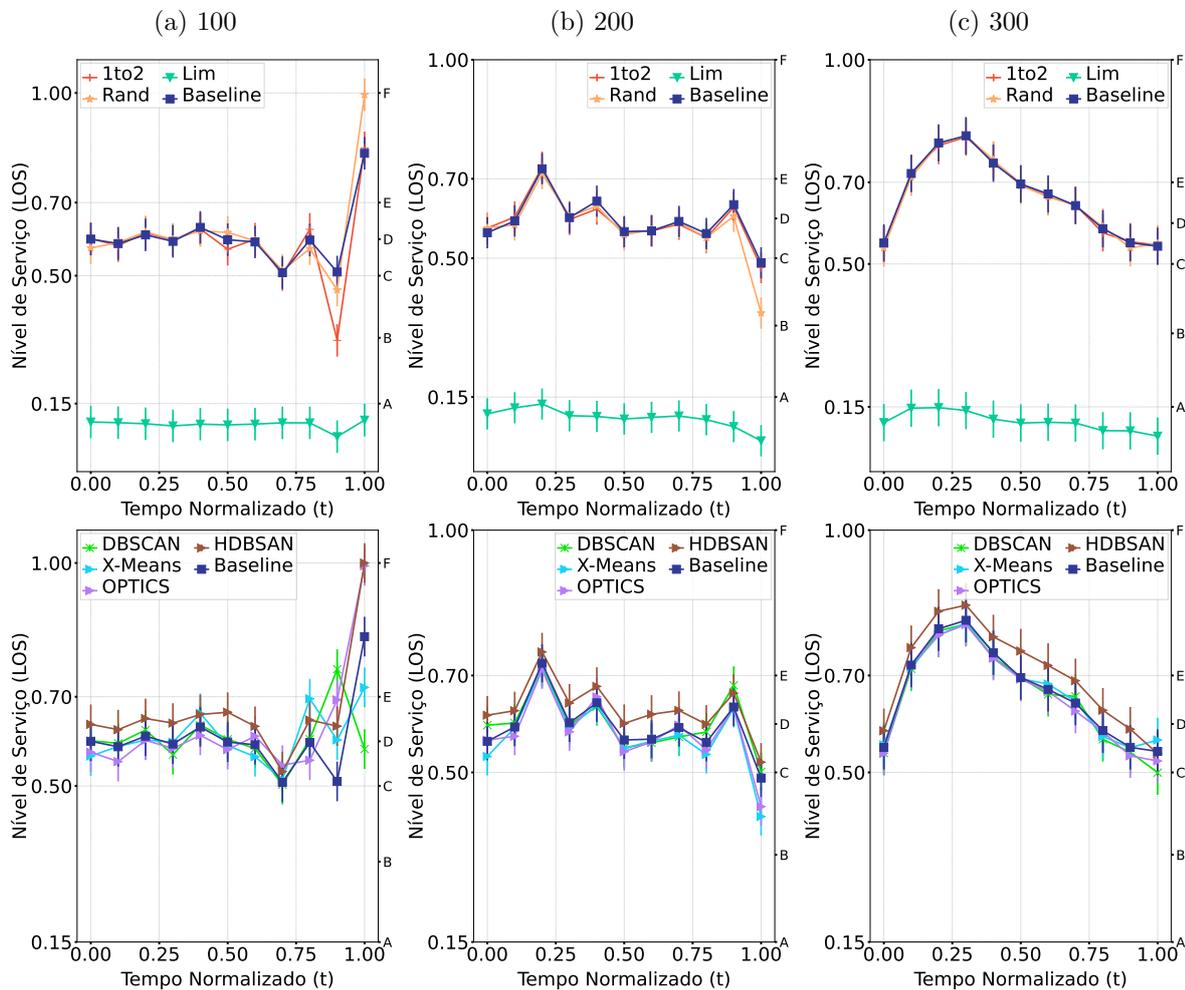


Figura 5.6: Análise do Level of Service (LOS) no Segmento (1)

A classificação por nível de serviço (LOS) permite definir, a partir de um modelo de classificação, o estado do fluxo de tráfego em cada segmento da rodovia. Assim, conforme já mencionado, esta análise é capaz de categorizar o nível de fluxo de tráfego com base

em fatores como velocidade, densidade e atraso de veículos durante as viagens realizadas ao longo das rodovias.

Analisando a Figura 5.6, pode-se observar, inicialmente, que há ligeira melhoria na acurácia, conforme a densidade veicular é aumentada. Observando-se a Figura 5.6a, é possível notar relativa perda de acurácia entre os algoritmos 1to2 e Random, especialmente nos minutos finais do experimento, no entanto, ainda que estes algoritmos demonstrem leve perda na acurácia, as variações identificadas não modificaram de forma significativa a classificação LOS, representada pelo segundo eixo  $y$  do gráfico.

Entre algoritmos baseados em agrupamento, esse efeito torna-se mais representativo, uma vez que a redução no conjunto de dados analisados pode impactar diretamente na qualidade da amostra selecionada e na consequente representatividade dos dados frente a um conjunto original. Este efeito pode ser observado ao analisar a Figura 5.6, na qual se verifica, entre os algoritmos de agrupamento, uma relativa perda de acurácia nos últimos minutos da execução. Este fenômeno pode ser explicado pela redução na quantidade de dados disponíveis para análise nos momentos finais do experimento.

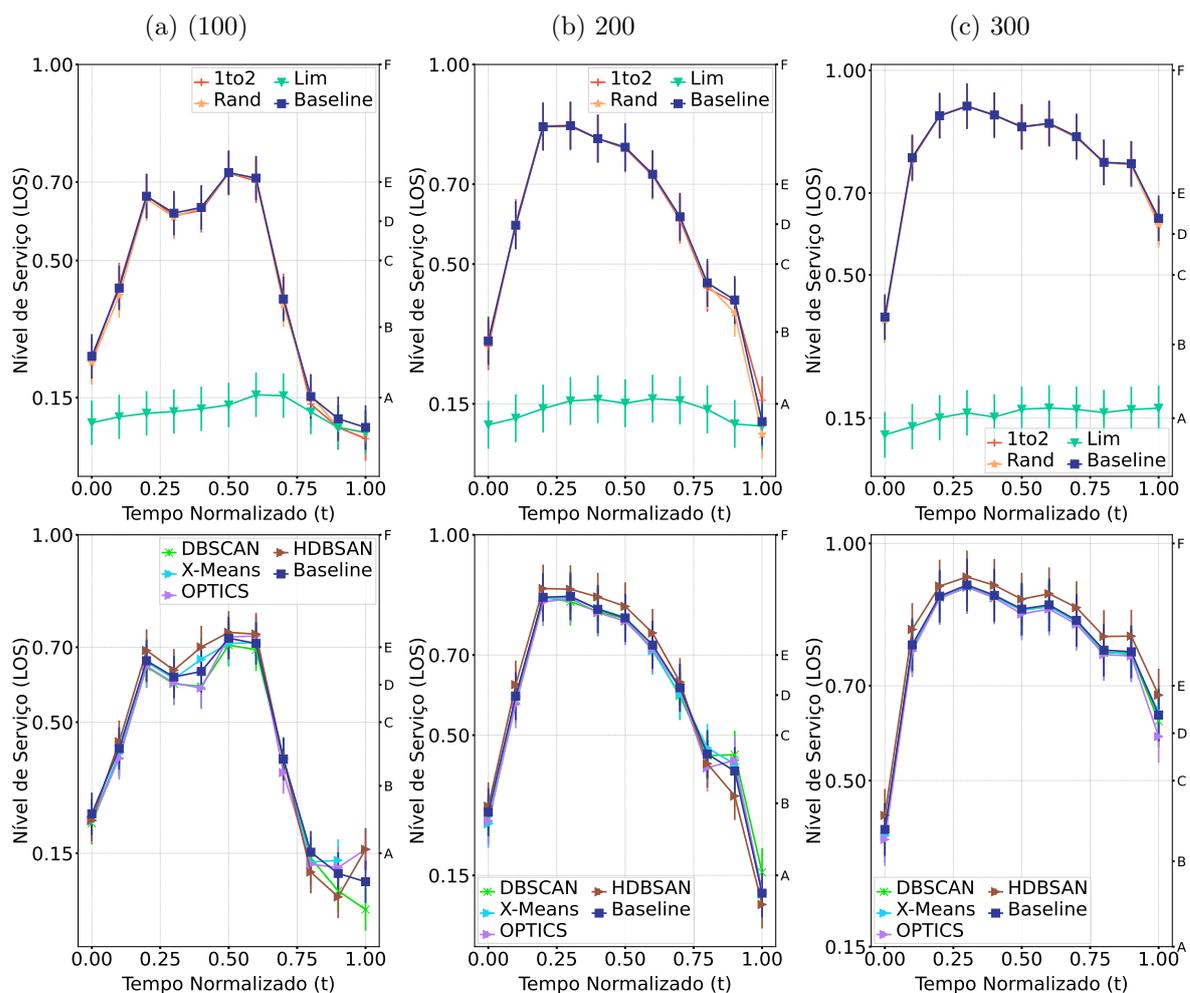


Figura 5.7: Análise do Level of Service (LOS) no Segmento (2)

Cabe observar ainda, que embora seja identificada essa relativa perda de acurácia nos momentos finais da execução, durante o curso do experimento, os algoritmos de agrupamento mantiveram um acompanhamento justo com as tendências observadas, mantendo-se, na maior parte do tempo, na mesma escala LOS observada no algoritmo *Baseline*. Este mesmo comportamento pode ser observado na Figura 5.7.

Analisando a Figura 5.7, é possível observar uma grande variação na classificação LOS, que se inicia com trânsito livre e, em seguida, registra um significativo aumento na densidade veicular. Esse comportamento está relacionado a formação de congestionamento e pode ser explicado pelo fato de que no segmento 2, o fluxo veicular converge para um ponto central, causando certa retenção em função da quantidade de veículos no segmento. Todos os algoritmos mantiveram níveis de acurácia satisfatórios. Cabe observar ainda, que o algoritmo HDBSCAN manteve um desempenho superior ao observado no segmento 1. Este comportamento pode ser explicado pelo fato da ocorrência e congestionamento garantirem um aumento na quantidade de dados coletada em função do tempo de permanência dos veículos na rodovia, permitindo a formação de novos *clusters* e garantindo uma seleção mais representativa.

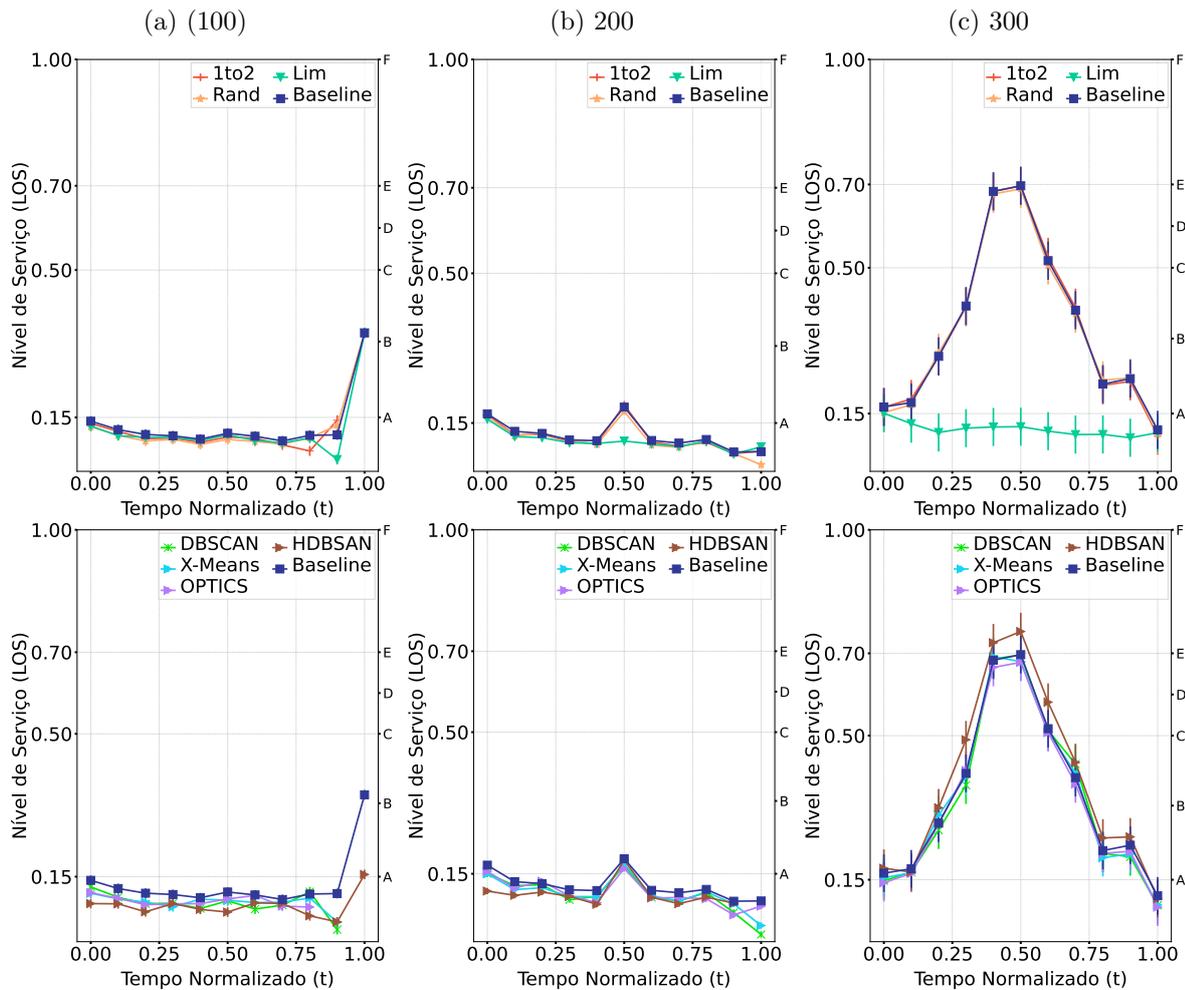


Figura 5.8: Análise do Level of Service (LOS) no Segmento (3)

Por fim, a análise do Segmento 3, ilustrado na Figura 5.8, evidencia um estado de trânsito livre para as densidades aferidas utilizando entre 100 e 200 veículos. No entanto, ao aumentar a densidade veicular para 300, observa-se uma mudança significativa no comportamento dos dados. Este fenômeno sugere que a capacidade máxima da rodovia foi alcançada, gerando a retenção ao longo do experimento. Todos os algoritmos mantiveram alinhamento com o *Baseline*. Entre os algoritmos de agrupamento, os algoritmos DBSCAN e XMEANS parecem ter tido um desempenho ligeiramente melhor. O algoritmo HDBSCAN identificou todas as mudanças de tendência e manteve-se dentro da escala de classificação LOS durante todo o curso do experimento.

### 5.3 APLICANDO A CLASSIFICAÇÃO POR NÍVEL DE SERVIÇO (LOS) EM TODO O CENÁRIO

Conforme mencionado, este trabalho enfatiza, entre outros aspectos, o monitoramento e detecção de congestionamento de tráfego através de um *framework* de redução de dados na névoa que utiliza algoritmos de agrupamento para tratamento de dados em fluxo contínuo obtido dos ambientes de mobilidade urbana.

Os experimentos apresentados nesta seção têm por fim mensurar a acurácia das técnicas de classificação, não só para posicionar a presente proposta frente a outros estudos analisados, mas também para estabelecer uma base comparativa confiável para o uso destas técnicas em diferentes cenários da computação urbana.

Para mensurar os níveis de acurácia obtidos e estabelecer padrões comparativos confiáveis foi utilizado o modelo de classificação por nível de serviço (*Level of Service* LOS). Conforme mencionado anteriormente, esta abordagem permite definir, a partir de um modelo de classificação, o estado do fluxo de tráfego em cada segmento da rodovia. Assim, esta análise é capaz de categorizar o nível de fluxo de tráfego com base em fatores como velocidade, densidade e atraso de veículos durante as viagens realizadas ao longo das rodovias. Consoante explicado na seção anterior, o padrão de classificação LOS compreende seis níveis de serviço mais comumente reconhecidos. A tabela 5.2 mostra esta classificação contendo as referências de cores utilizadas nas visualizações a seguir.

Tabela 5.2: Classificação ( $w_{ij}$ ) das Condições de Tráfego

Peso da Rodovia ( $w_{ij}$ )	LOS	Fluxo de Tráfego
[0, 0.15]	A ●	Free-flow
[0.15, 0.33]	B ●	Reasonably free-flow
[0.33, 0.50]	C ●	Stable-flow
[0.50, 0.60]	D ●	Approaching unstable-flow
[0.60, 0.70]	E ●	Unstable-flow
[0.70, 1.00]	F ●	Breakdown-flow

Com o fim de observar o comportamento do tráfego na perspectiva do modelo de classificação por níveis de serviço LOS, aplicado no contexto de fluxo de dados contínuo, será apresentado, a partir deste ponto, um conjunto de experimentos nos quais os mapas são populados com diferentes densidades de veículos. Nestes experimentos, as densidades variam entre 100, 200 e 300 veículos, respectivamente.

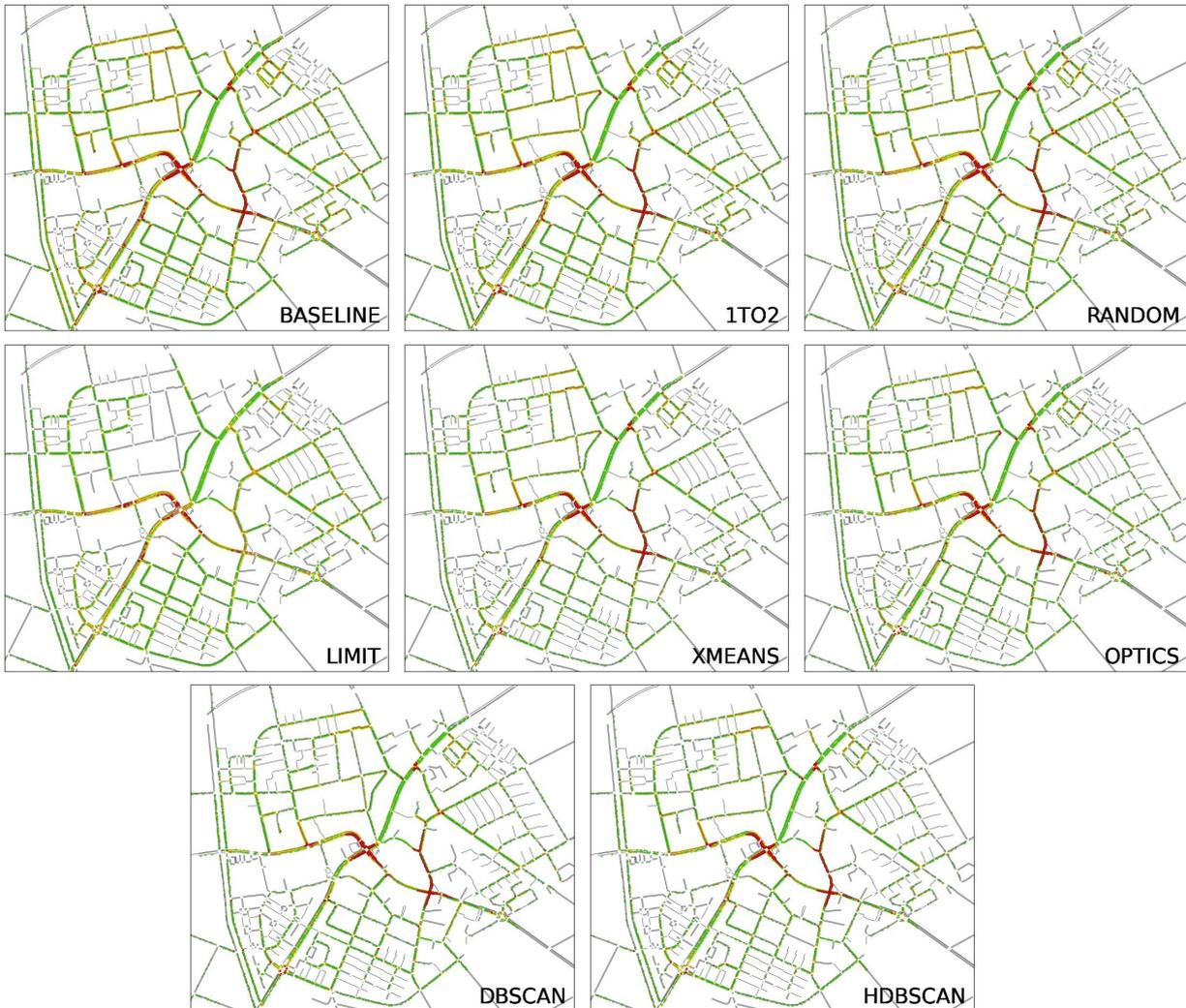


Figura 5.9: Detecção de congestionamento de tráfego com densidade de 100 veículos/ $km^2$

O primeiro objeto de análise é ilustrado na Figura 5.9. Na imagem do mapa é possível observar os resultados da aplicação da abordagem de classificação por nível de serviço LOS utilizando uma densidade de 100 veículos no cenário. Neste contexto, observa-se que a aplicação da métrica LOS nos algoritmos *Baseline*, *1to2* e *Rand* mantém comportamentos semelhantes, demonstrando, na maioria dos casos, similaridade entre os pontos, tendências, além de manter os mesmos níveis de identificação de congestionamento de tráfego. Embora o DBSCAN não identifique todas as variações de densidade, em função da baixa densidade de veículos neste cenário, ele é capaz de identificar a maior parte dos pontos de congestionamento formados nas rodovias analisadas. Este mesmo comporta-

### 5.3 APLICANDO A CLASSIFICAÇÃO POR NÍVEL DE SERVIÇO (LOS) EM TODO O CENÁRIO091

mento pode ser observado nos algoritmos OPTICS e XMEANS. O algoritmo HDBSCAN ilustra uma leve perda de acurácia que será analisada de forma mais detalhada na seção seguinte.

Já a Figura 5.10 ilustra o comportamento dos algoritmos quando a densidade veicular é aumentada para 200 veículos por  $km^2$ . Nesse novo cenário, enquanto o algoritmo Lim mantém uma capacidade de detecção de congestionamento inferior aos demais algoritmos, pode-se observar um certo aumento na acurácia no que refere a detecção de pontos de retenção localizados pelo algoritmo de agrupamento, em geral. Esta melhoria se deve ao aumento na quantidade de dados, o que torna a amostra mais representativa em relação aos experimentos envolvendo 100 veículos, analisados na Figura 5.9. Este aumento na precisão é observado também nos resultados do algoritmo DBSCAN pelas mesmas razões.

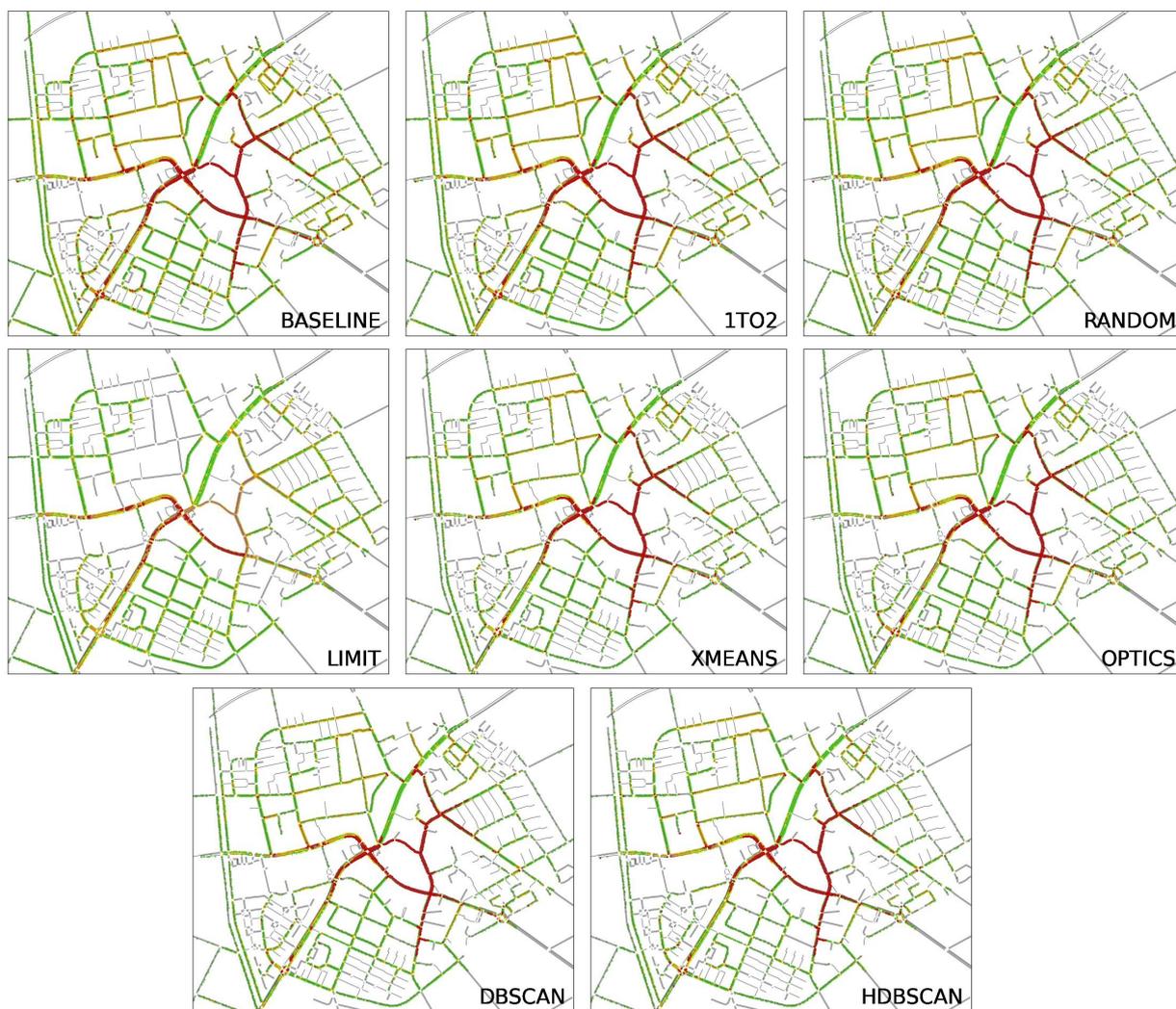


Figura 5.10: Detecção de congestionamento de tráfego com densidade de 200 veículos/ $km^2$

Quando o aumento do fluxo de tráfego local é ampliado para trezentos veículos por  $km^2$ , como pode ser visto na Figura 5.11, pode-se notar maior formação de congestionamento e, como consequência, maior presença dos pesos E e F, de acordo com a abordagem

de classificação por nível de serviço LOS. Esta classificação pode ser vista na tabela 5.2.

Nesse cenário, o algoritmo Lim demonstrou uma ligeira melhora na detecção de congestionamento do que os cenários anteriores. Este aumento de desempenho pode sugerir que a aplicação dessa técnica pode ser mais eficiente em situações de fluxo de tráfego com maior concentração de veículos. Entre os algoritmos de agrupamento, embora em todos os casos os algoritmos mantiveram uma significativa representação do conjunto de dados original ilustrado pelo *Baseline*, é possível observar uma leve melhora no algoritmo HDBSCAN. Já o algoritmo DBSCAN manteve níveis de precisão mais acurados durante todas as variações de densidade, demonstrando com precisão os pontos de retenção e os níveis de congestionamento das rodovias.

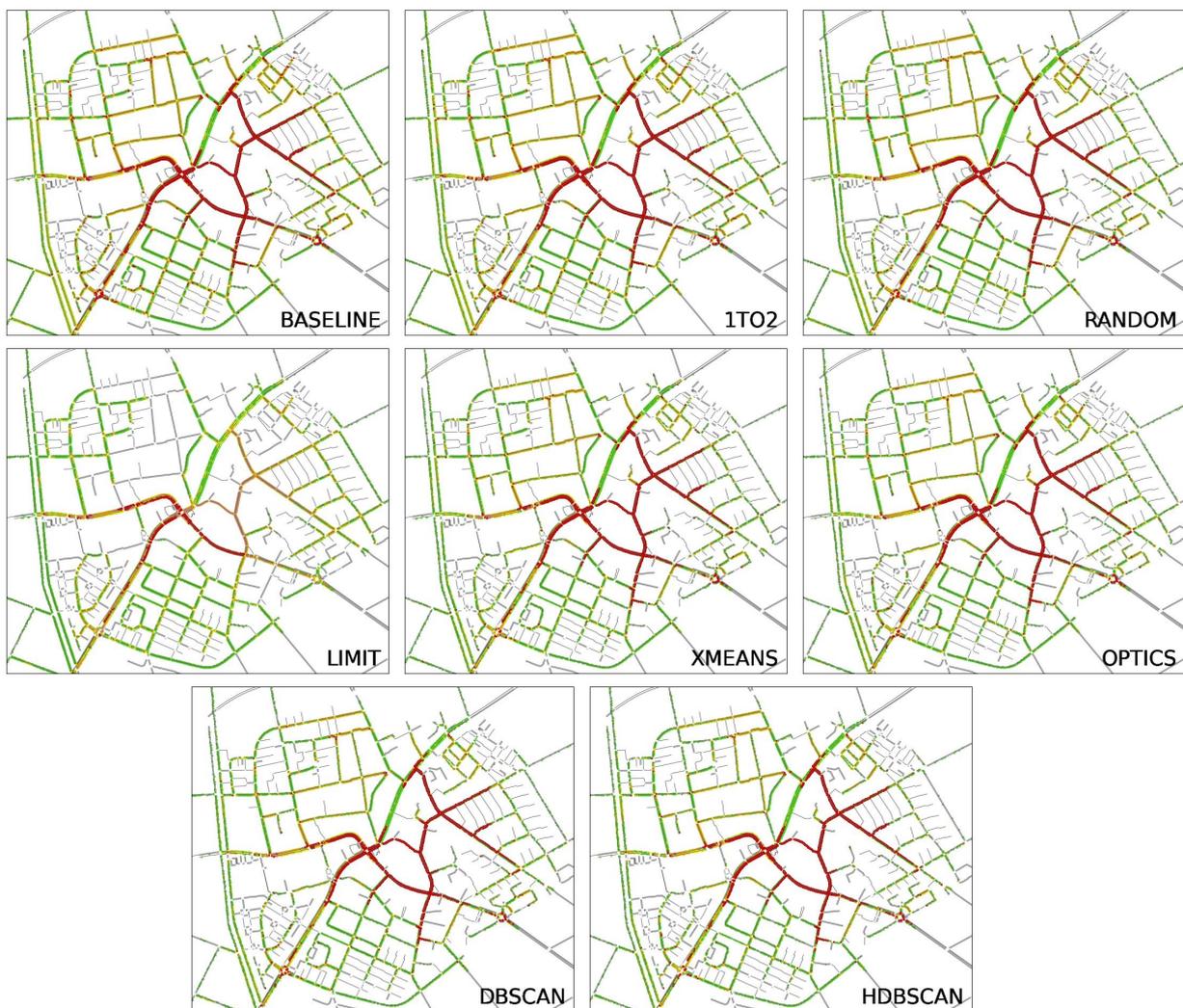


Figura 5.11: Detecção de congestionamento de tráfego com densidade de 300 veículos/ $km^2$

Analisando as três Figuras 5.9, 5.10 e 5.11 ,concomitantemente, pode-se observar que o algoritmo Lim manteve menores níveis de representatividade em relação ao nível de serviço LOS do que os demais. Isso pode ser explicado pela baixa aquisição de dados de

tráfego. Esta baixa representatividade frente ao conjunto de dados original prejudica a aplicação adequada da análise, classificação e uso deste conjunto de dados na tomada de decisão.

Por outro lado, o algoritmo X-Means que se utiliza de uma estratégia de agrupamento baseada em similaridades, desconsiderando, em certa medida, as questões relativas ao posicionamento físico dos pontos, manteve significativa representatividade ao longo de todos os experimentos, mantendo-se similar aos algoritmos DBSCAN e OPTICS e mantendo níveis de acurácia superiores ao HDBSCAN.

Adicionalmente, cabe lembrar que este trabalho preconiza, entre outros aspectos, o uso de algoritmos de agrupamento como forma de garantir a redução de dados, no entanto, esse conjunto de dados resultante deve manter níveis de acurácia satisfatórios, capazes de representar um certo conjunto de dados original. Assim, a próxima seção apresentará uma análise dos níveis de acurácia das visualizações apresentadas, baseada no erro quadrático médio ou *root-mean-square deviation* (RMSE).

#### 5.4 ANÁLISE DA ACURÁCIA

Ao analisar os algoritmos aplicados à detecção de congestionamento ilustrados na seção anterior, foi possível observar grande similaridade no desempenho dos algoritmos, destacando-se apenas ligeiras variações frente ao *Baseline*. Diante disso, a fim de mensurar de forma mais consistente os níveis de acurácia identificados nestas visualizações, a Figura 5.12 destaca a aplicação RMSE (*Root-mean-square deviation*) para ilustrar o comportamento dos algoritmos na representação do conjunto de dados original.

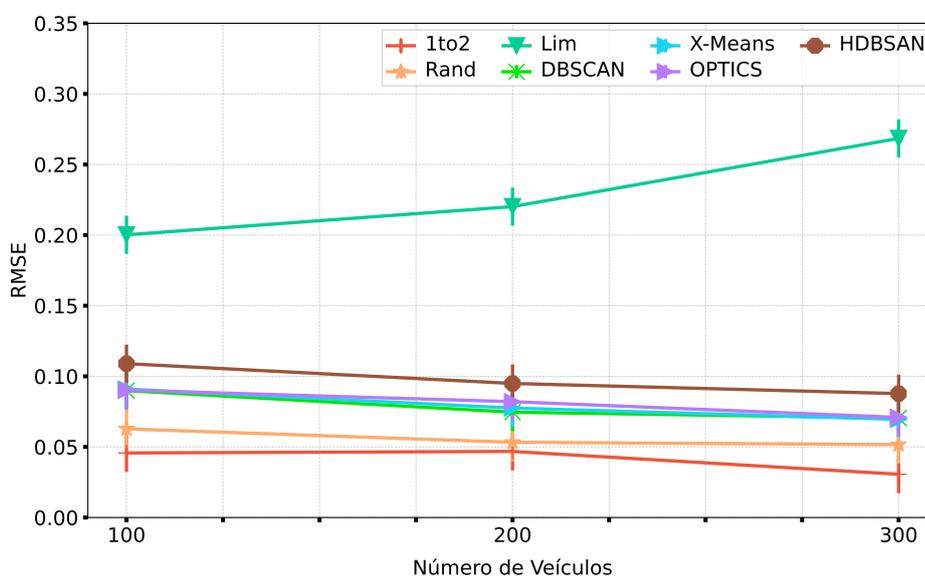


Figura 5.12: Aplicação do *Root-mean-square deviation* (RMSE) para as densidades de 100, 200 e 300 veículos

A métrica *root mean squared error* ou RMSE foi aplicada sob os conjuntos de dados processados como forma de mensurar os níveis de acurácia obtidos na aplicação de cada

um dos algoritmos durante o processo de detecção de congestionamento de tráfego.

Analisando a Figura 5.12, pode-se notar que os algoritmos 1to2 e Random mantiveram níveis de acurácia superiores, ou seja, incorreram em um erro médio menor do que os demais. Esta realidade pôde ser observada também durante a análise das visualizações dos mapas. No tocante aos algoritmos baseados em agrupamento, observa-se que o algoritmo DBSCAN tem um desempenho ligeiramente melhor que os demais, seguido de perto pelos algoritmos OPTICS e X-Means. O algoritmo HDBSCAN manteve níveis de acurácia satisfatórios, mas ligeiramente menores do que os demais. O algoritmo Lim teve desempenho inferior aos demais algoritmos. Conforme descrito anteriormente, este efeito pode ser explicado pela estratégia de definição do limite que desconsidera o contexto particular de cada rodovia.

Cabe observar ainda que, de forma similar as análises evidenciadas pela classificação LOS, aferidas nas Figuras 5.6, 5.7 e 5.8, o erro médio reduz a medida que a quantidade de dados disponíveis aumenta em função do aumento da densidade veicular. Esse fenômeno torna-se mais perceptível entre os algoritmos de agrupamento, em função da própria natureza e estratégia de seleção de dados adotadas durante o processamento.

A fim de oferecer uma análise mais detalhada acerca do comportamento dos dados processados frente a um conjunto de dados original, propõe-se a seguir a análise individual dos algoritmos utilizando uma abordagem baseada em regressão linear, variando-se as densidades entre 100, 200 e 300 veículos. A Figura 5.13 ilustra este cenário para uma densidade de 100 veículos/ $km^2$ .

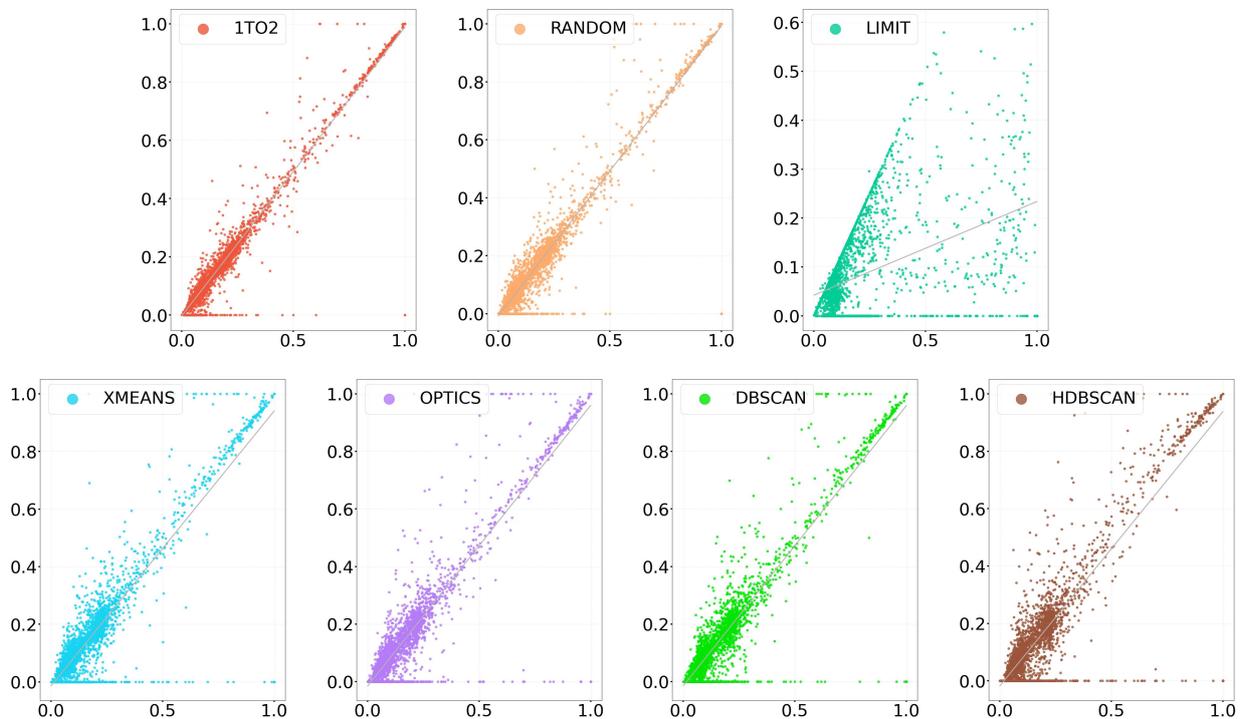


Figura 5.13: Análise da dispersão dos dados processados aplicando uma densidade de 100 veículos/ $km^2$

Analisando a Figura 5.13, pode-se observar que, na maioria dos casos, há uma correlação mais significativa entre os valores mais baixos dos gráficos. Entre os algoritmos baseados em amostragem simples como 1to2 e Random, esse comportamento, apesar de presente, torna-se mais atenuado na medida que os valores se tornam mais distribuídos em função da reta normal. Entre os algoritmos baseados em agrupamento pode-se observar que a dispersão aumenta progressivamente até um certo limite do conjunto de dados.

Esse fenômeno pode ser explicado pelo contínuo processo de formação de *clusters* e estratégia de seleção de dados. Certos elementos contidos em um *cluster* com menor representatividade tornam-se residuais nessa visualização. Ao direcionar as análises apenas para os algoritmos de agrupamento, pode-se observar que essa dispersão é mais acentuada no algoritmo HDBSCAN que manteve um desempenho ligeiramente inferior ao OPTICS e XMEANS. Os dados processados pelo algoritmo Lim demonstraram baixa correlação frente ao conjunto de dados original.

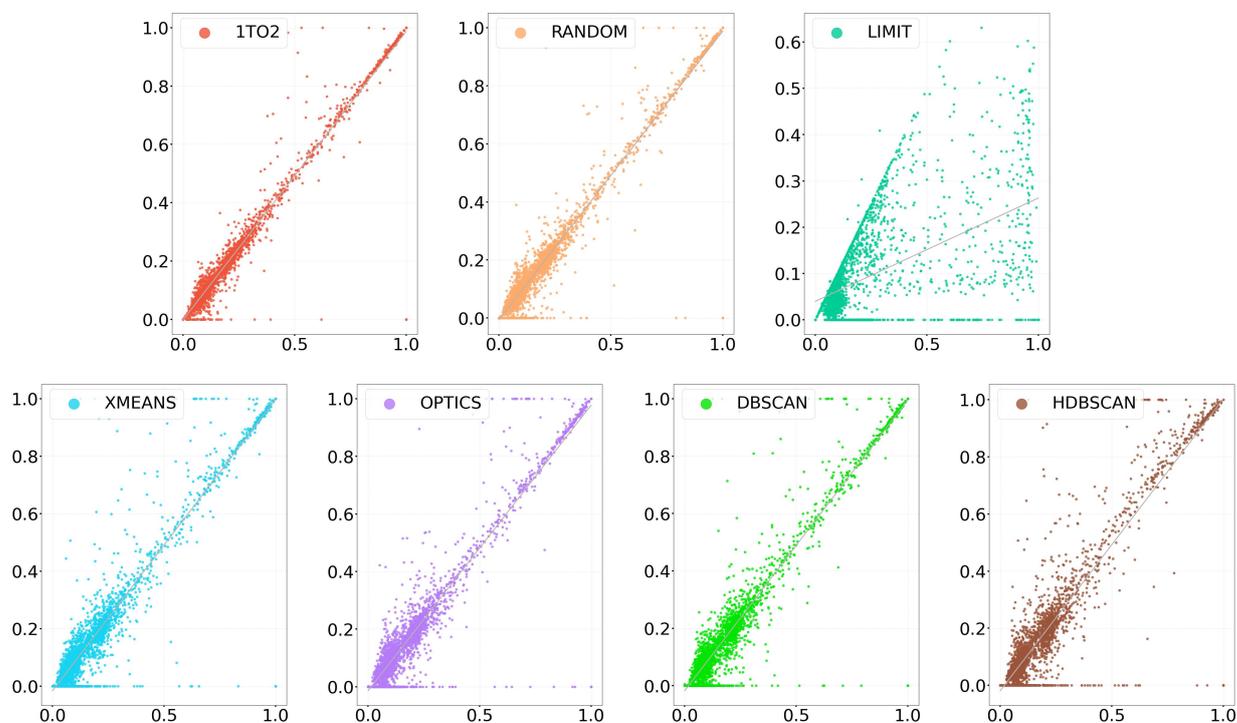


Figura 5.14: Análise da dispersão dos dados processados aplicando uma densidade de 200 veículos/ $km^2$

Ao aumentar a densidade para 200 veículos/ $km^2$ , conforme ilustrado na Figura 5.14, pode-se observar um aumento na correção geral dos dados processados. Para os algoritmos 1to2 e Rand pode-se notar a redução dos pontos residuais e uma maior concentração de pontos para valores mais altos. Este mesmo comportamento pode ser observado entre os algoritmos baseados em agrupamento, onde o algoritmo DBSCAN evidencia uma ligeira melhora frente aos demais.

Analisando a Figura 5.15, quando a densidade veicular é ampliada para 300 veículos/ $km^2$

observa-se um esperado aumento da correção dos dados, no entanto, cabe destacar o significativo aumento dessa correlação entre os valores médios dos eixos. Para os algoritmos baseados em amostragem simples como 1to2 e Random, pode-se inferir uma correlação direta entre a quantidade de dados e os níveis de representatividade.

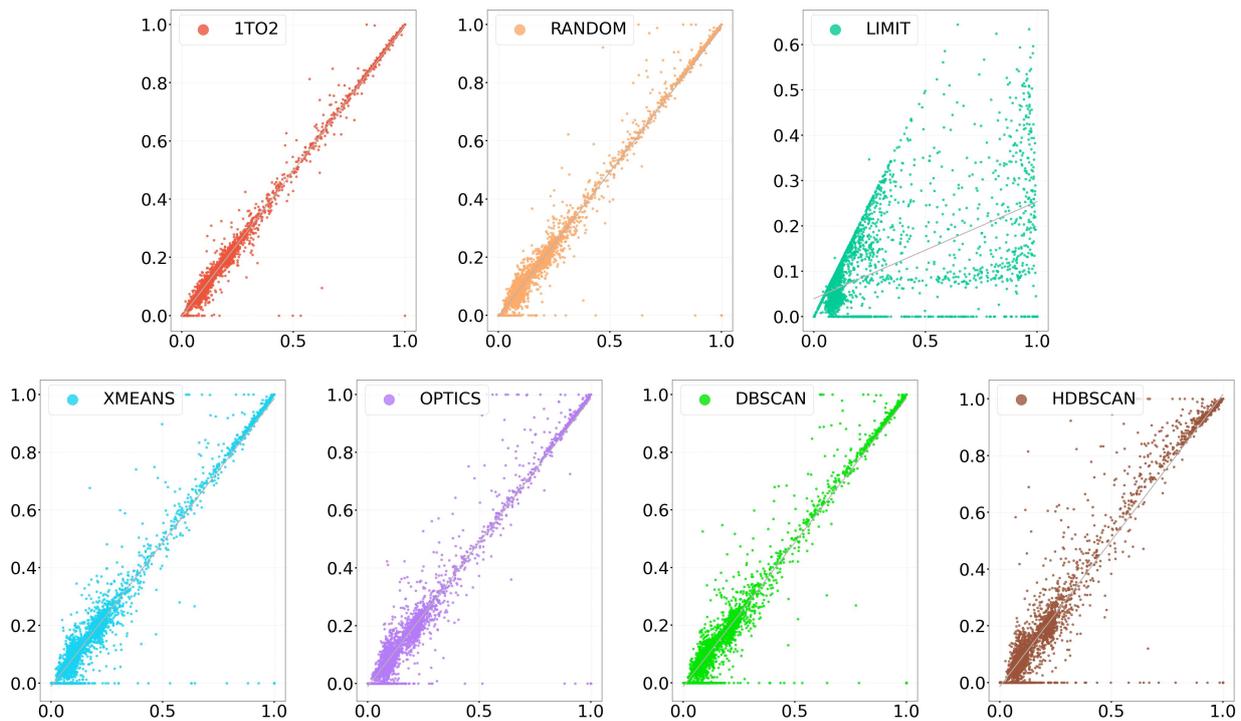


Figura 5.15: Análise da dispersão dos dados processados aplicando uma densidade de 300 veículos/ $km^2$

Para os algoritmos baseados em agrupamento, esse comportamento pode ser explicado pela melhoria na qualidade dos *clusters* formados. O aumento da densidade veicular amplia o processamento do fluxo de dados contínuo, resultando no aumento dos dados submetidos a processamento pelos algoritmos. Essa dinâmica favorece a seleção de dados com maior representação do conjunto original. O algoritmo Lim manteve-se com níveis de correlação mais baixos dos que observados nos experimentos anteriores.

Através da análise do erro médio, a partir de um modelo de regressão para cada um dos algoritmos, foi possível identificar seus respectivos comportamentos em função da quantidade de ruídos observados, no entanto, um outro aspecto a ser analisado consiste na qualidade dos *clusters* formados, tende-se a acreditar que estes níveis de qualidade podem ter impacto direto nos níveis de acurácia alcançados. Desde modo, a seção seguinte avalia o *coeficiente de silhueta* (CS(i)) dos algoritmos analisados, a fim de identificar o comportamento e o processo de formação de *cluster* ao longo dos experimentos.

## 5.5 COEFICIENTE DE SILHUETA

O coeficiente de silhueta consiste em uma medida de qualidade com o fim de verificar a estrutura dos *clusters* resultantes do processamento de algoritmos baseados em agrupamento. No contexto deste trabalho, essa medida foi utilizada para permitir uma avaliação sistemática dos valores resultantes do processamento dos algoritmos X-Means, OPTICS, DBSCAN e HDBSCAN, variando as densidades entre 100, 200 e 300 veículos/ $km^2$  durante o ciclo de replicações do experimento.

O cálculo do coeficiente de silhueta consiste em quantificar tanto a *coesão* quanto a *separação* entre grupos de instâncias  $n$ . A coesão  $a_{xi}$  de uma instância  $xi$  é calculada como a distância média entre  $xi$  e todas as outras instâncias no mesmo grupo que  $xi$ . A separação  $b_{xi}$  é a distância mínima entre  $xi$  e instâncias em todos os outros grupos. Rousseeuw (ROUSSEEUW, 1987) denota o Coeficiente de Silhueta por Equação 5.1

$$\frac{1}{n} \sum_{xi \in \mathcal{X}} \frac{(b_{xi} - a_{xi})}{\max(a_{xi}, b_{xi})} \quad (5.1)$$

O coeficiente da silhueta mantém-se entre um intervalo de valores que variam entre  $[-1; 1]$ , nesse contexto, valores menores sugerem menor coesão e separação entre *clusters*, resultando em uma seleção de menor qualidade. Da mesma forma, valores maiores sugerem a formação de *clusters* mais coesos, aumentando as chances de obter amostras mais representativas frente a um certo conjunto de dados original.

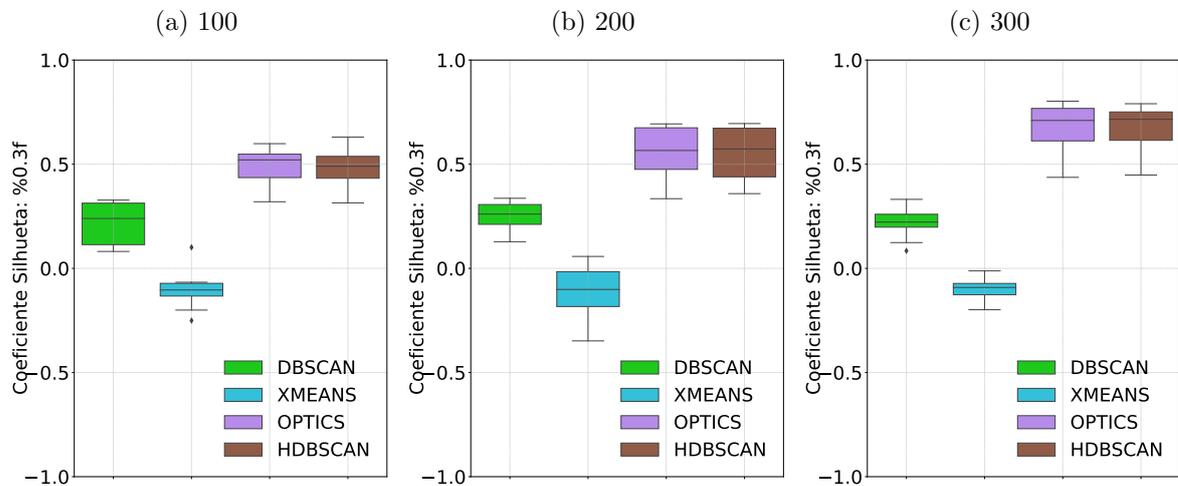


Figura 5.16: Análise do Coeficiente Silhueta para 100, 200 e 300 veículos/ $km^2$

A Figura 5.16 mostra o resultado do coeficiente da silhueta para os algoritmos de agrupamento utilizados neste trabalho, variando a densidade veicular, conforme mencionado anteriormente. Analisando as variações ocorridas frente ao aumento da densidade veicular, pode-se observar um desempenho similar entre os algoritmos OPTICS e HDBSCAN, ambos os algoritmos apresentam uma variabilidade maior no cenário com 200 veículos. Entre os algoritmos DBSCAN e X-Means, pode-se observar que a variabilidade reduz a

medida que a densidade veicular aumenta, este fenômeno pode ser explicado pela capacidade do algoritmo de lidar com o processamento do fluxo de dados, considerando os aspectos pertinentes a localização, direção e rodovias. O algoritmo X-Means demonstrou índices menores relativos ao cálculo do coeficiente.

A fim de observar o comportamento dos algoritmos ao longo de cada experimento, a Figura 5.17 ilustra o cálculo do coeficiente da silhueta em função do tempo de simulação. Nesta visualização, o tempo foi normalizado, a fim de facilitar a análise dos dados. Além disso, um segundo eixo  $y$  foi configurado como forma de evidenciar a relação entre o cálculo deste coeficiente e a quantidade dos *clusters* formados ao longo do experimento. Assim como nas demais análises, cada conjunto de experimentos foi projetado para as densidades 100, 200 e 300 veículos/ $km^2$ .

Em um cenário com 100 veículos, pode-se observar que os algoritmos HDBSCAN e OPTICS mantiveram índices mais altos do que os algoritmos DBSCAN e X-Means. Importante notar que embora o algoritmo HDBSCAN tenha utilizado uma quantidade ligeiramente menor de *clusters* do que o algoritmo OPTICS ao longo da simulação, o comportamento deles manteve-se similar. Quanto ao algoritmo DBSCAN, observa-se que, em uma escala menor, houve certa variabilidade no processo de criação de *clusters*, no entanto, o coeficiente manteve-se estável ao longo de toda a simulação. O algoritmo X-Means apresentou menor coesão entre os algoritmos analisados.

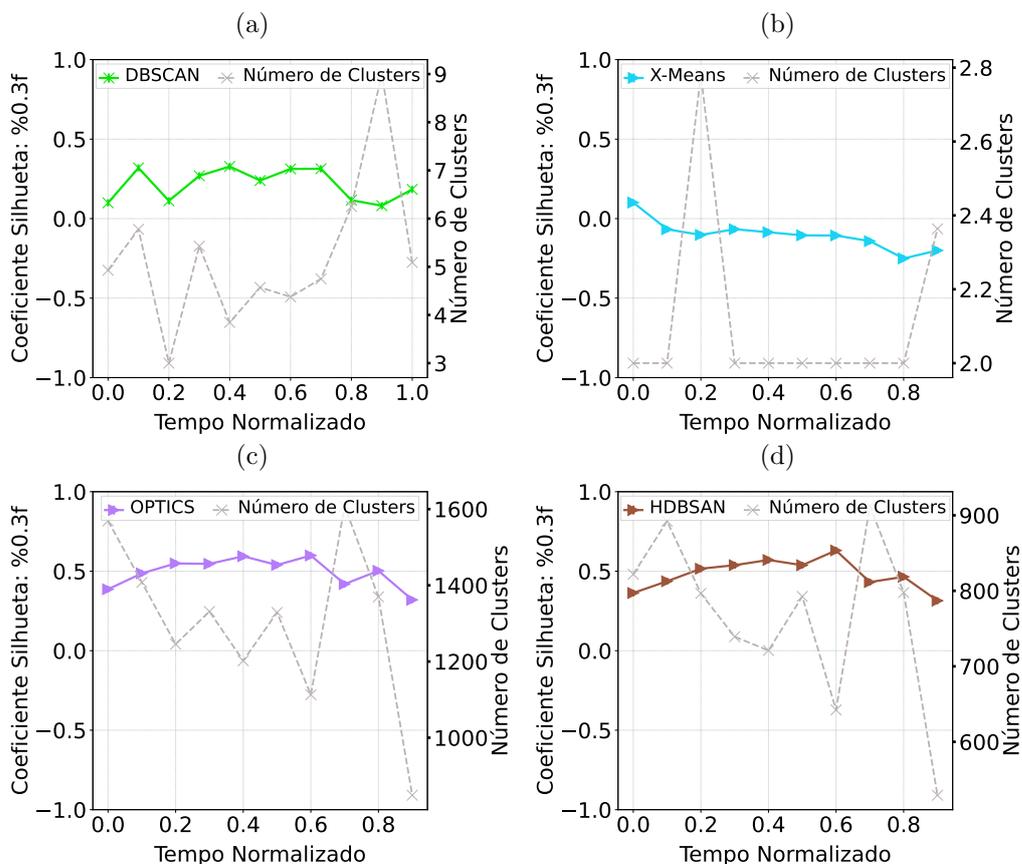


Figura 5.17: Coeficiente silhueta x *clusters* formados em um cenário com 100 veículos

No cenário contendo 200 veículos, ilustrado na Figura 5.18, pode-se observar um aumento do coeficiente entre os algoritmos HDBSCAN e OPTICS, além disso, pode-se observar ainda uma variabilidade maior do algoritmo X-Means. O algoritmo DBSCAN manteve-se estável. No que se refere a formação de *clusters*, pode-se observar um aumento significativo na quantidade de *clusters* para o algoritmo X-Means em relação ao experimento anterior. Este comportamento pode explicar a redução na aferição do coeficiente silhueta para este algoritmo.

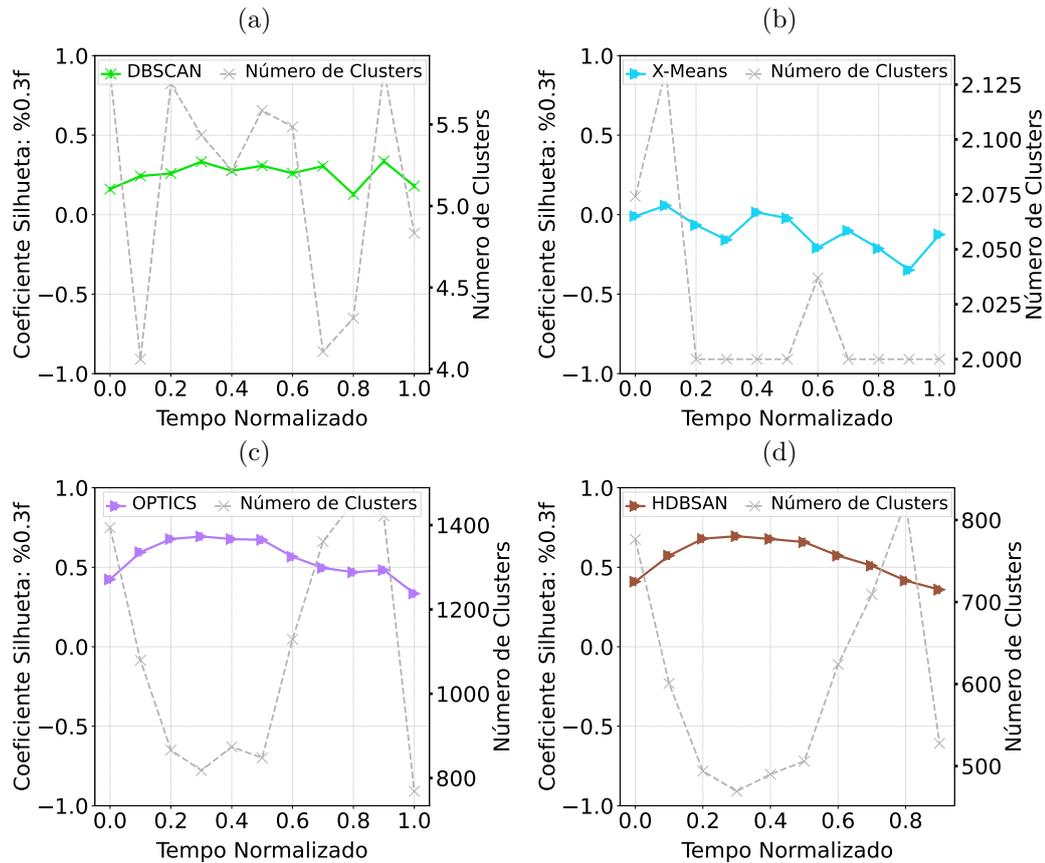


Figura 5.18: Coeficiente silhueta x clusters formados em um cenário com 200 veículos

Por fim, analisando a Figura 5.19, em um cenário com 300 veículos, pode-se perceber um relativo aumento dos resultados do coeficiente da silhueta entre os algoritmos HDBSCAN e OPTICS. Além disso, pode-se notar ainda que em ambos os algoritmos há uma redução progressiva da quantidade de *clusters* formados. Esse fenômeno pode ser explicado pelo fato de que neste cenário a incidência de congestionamento torna-se significativamente maior, fazendo com que os grupos se tornem mais coesos e permaneçam no cenário por mais tempo.

Quanto ao algoritmo DBSCAN, é possível notar uma redução na variação dos valores de coeficiente frente aos experimentos anteriores, além disso, pode-se observar que de forma contrária aos algoritmos OPTICS e HDBSCAN, o algoritmo DBSCAN demonstrou um ligeiro aumento na quantidade de *clusters*. Este comportamento pode estar relacionado ao fato de que o algoritmo DBSCAN utiliza o posicionamento dos pontos como

forma de identificar variações nas densidades. Esta característica associada ao fato de que os veículos estão distribuídos nas ruas e slots por mais tempo, podem justificar o aumento na formação de *clusters*. O algoritmo X-Means apresentou relativa melhora e uma redução significativa na variação dos valores de coeficiente aferidos ao longo do experimento.

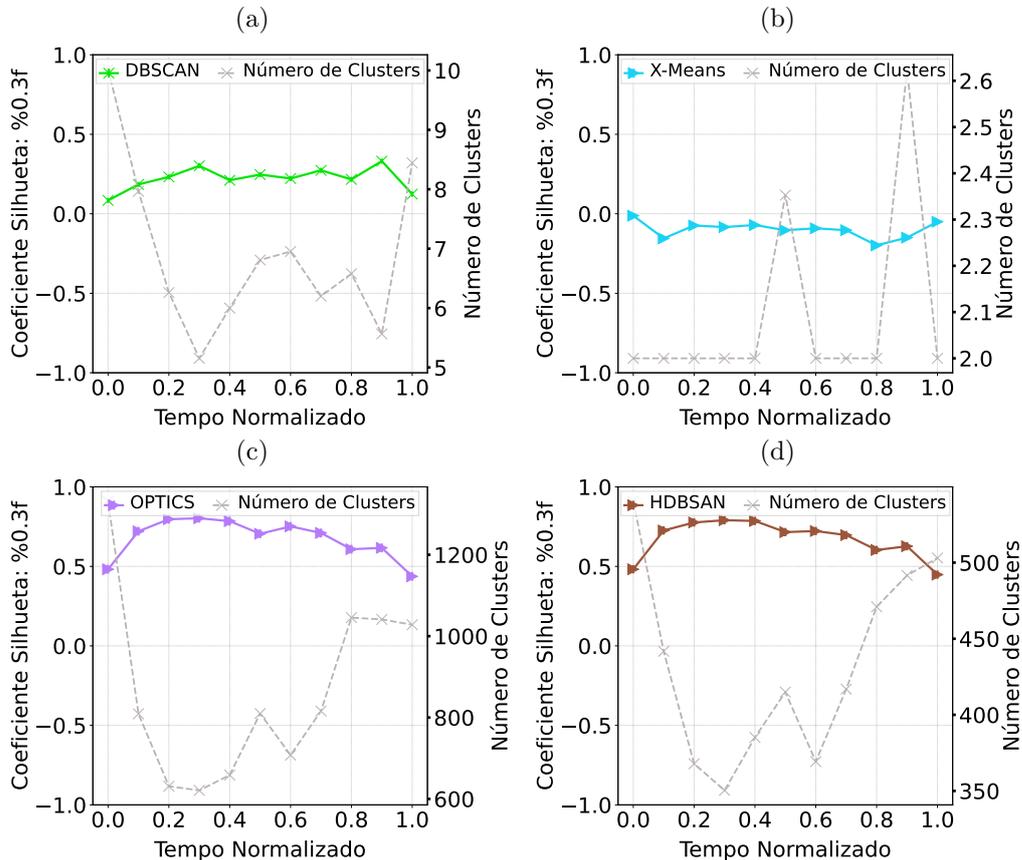


Figura 5.19: Coeficiente silhueta x clusters formados em um cenário com 300 veículos

O coeficiente da silhueta é uma das métricas mais importantes para representar a qualidade para algoritmos de agrupamento, em geral. No entanto, em um cenário de computação urbana, esta métrica pode apresentar relativas distorções em função da natureza e da dinâmica do fluxo de veículos nas rodovias. Este fenômeno ocorre porque a abordagem desconsidera questões relacionadas ao tráfego, como a direção do veículo e a posição na via para a análise e avaliação dos grupos.

A qualidade dos *clusters* e a sua representatividade frente a um conjunto de dados original é fator preponderante para os processos de seleção e redução de dados propostos neste trabalho. A abordagem de redução de dados preconiza a seleção de um subconjunto representativo, capaz de concomitantemente, *a)* reduzir a carga de dados trafegados pela rede e ainda assim, *b)* manter níveis de acurácia aceitáveis para garantir o entendimento claro destes subconjuntos. Na próxima seção, será apresentado um conjunto de experimentos com o fim de mensurar os impactos da redução de dados no contexto do tráfego, uso e consumo dos recursos de rede.

## 5.6 ANALISANDO O USO DOS RECURSOS DE REDE

Nesta seção, serão analisadas questões ligadas ao desempenho das estratégias de redução de dados em um contexto de fluxo de dados contínuo, avaliando o consumo de recurso de rede durante os processos de monitoramento e detecção de congestionamento de tráfego.

### 5.6.1 Uso da Rede e Custos de Comunicação

Um dos principais objetivos deste trabalho consiste em estabelecer um modelo eficiente para redução de dados utilizando o paradigma da computação em névoa em um contexto de fluxo de dados contínuo. Nestes cenários, grandes volumes de dados são gerados pela camada de mobilidade e este comportamento pode levar ao congestionamento da rede, perda de pacotes, aumento dos custos de comunicação, desperdício de largura de banda e aumento da latência.

A largura de banda da rede ( $NB$ ) é dada por  $S_B/T$ , onde  $S_B$  representa os dados de tráfego e  $T$  é a largura de banda do canal.  $S_B$  também é dado por  $(L_B + \lambda_B)$ , onde  $L_B$  é a soma de todos os tamanhos de pacotes e  $\lambda_B$  é a taxa média de chegada. Quanto maior o número de veículos, maior é o  $NB$  consumido. Uma maneira de abordar este problema é minimizar  $L_B$  sujeito à restrição  $a_o \geq a_{min}$ , onde  $a_o$  é a precisão obtida e  $a_{min}$  é a precisão mínima aceitável para detectar congestionamento de tráfego. A Figura 5.20 ilustra os resultados do consumo da rede para cada um dos métodos utilizados ao longo dos experimentos. As análises consideraram as medidas de densidade variando entre 100, 200 e 300 veículos/ $km^2$ .

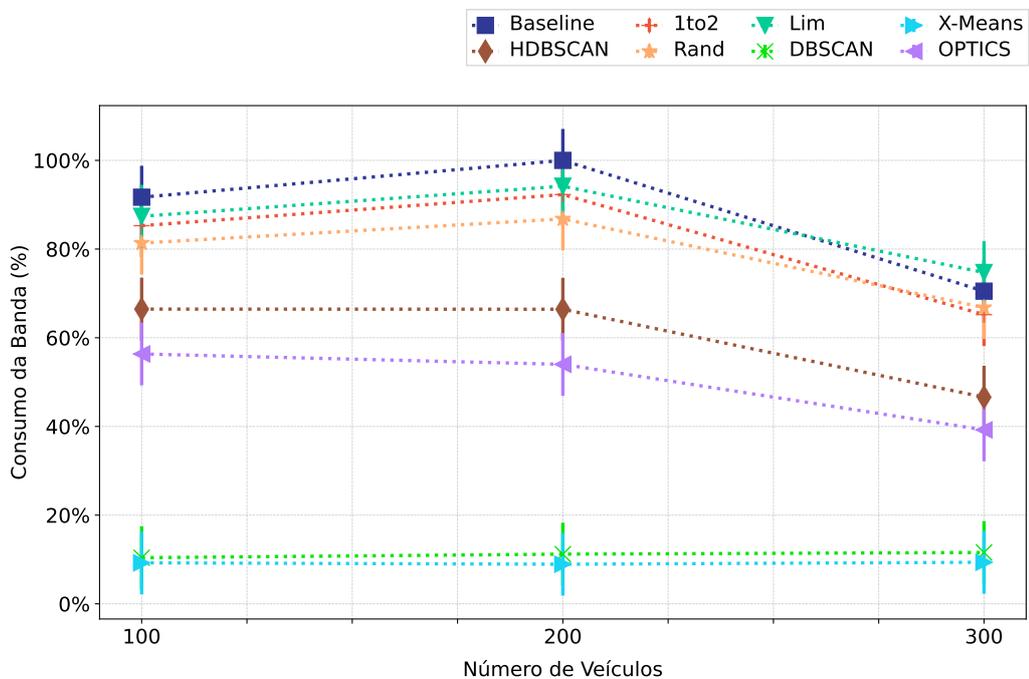


Figura 5.20: Comparativo da Largura de Banda e a Quantidade de Veículos.

Inicialmente, cabe observar que a eficiência dos resultados descritos na Figura 5.20 passa não só pela acurácia na identificação dos congestionamentos quando comparados aos resultados da estratégia que utiliza o algoritmo *Baseline*, mas também pelo comportamento destes algoritmos no que se refere ao uso de recurso de rede. Diante disso, observa-se que uma boa estratégia de redução de dados deve atuar nas duas frentes de forma concomitante, *a)* reduzindo os dados de forma significativa, ao passo que, *b)* mantendo níveis de acurácia suficientes para a detecção dos pontos de retenção das rodovias. Adicionalmente, estes conjuntos de dados resultantes devem ser capazes de fornecer insumos para suportar as decisões de intervenções ou re-roteamentos dos veículos na camada de mobilidade.

A Figura 5.20 ilustra os resultados de uso dos recursos de rede para os algoritmos: **Baseline**, **1to2**, **Lim**, **Rand**, **X-Means**, **OPTICS**, **DBSCAN** e **HDBSCAN** respectivamente. Nesse contexto, pode-se observar que os algoritmos de agrupamento são menos onerosos no que se refere ao consumo da rede. O algoritmo *Baseline* apresenta resultados de uso de rede mais elevados, uma vez que não há redução de dados nesta estratégia. Entre os algoritmos de redução de dados baseados em amostragem simples como 1to2, Rand e Lim, pode-se perceber valores médios próximos aos identificados no algoritmo *Baseline*. Este comportamento pode estar relacionado ao fato de que nestas estratégias, o mecanismo de envio de dados permanece similar ao adotado pelo algoritmo *Baseline*. A inexistência do conceito de janelas temporais nestes algoritmos, mantém o envio ininterrupto dos dados, conforme são recebidos da camada de mobilidade, gerando níveis similares de tráfego durante toda a simulação. Deste modo, se o uso do recurso de rede é alto, permanecerá alto ao longo de todo o experimento.

Por outro lado, os algoritmos OPTICS e HDBSCAN alcançaram menos de 55% do uso da rede para todos os cenários, além disso, mantiveram uma boa precisão na detecção por nível de serviço (LOS). Entretanto, cabe destacar que o algoritmo OPTICS apresentou um desempenho ligeiramente superior ao algoritmo HDBSCAN. No que se refere aos algoritmos DBSCAN e X-Means, ambos mantiveram um desempenho significativamente melhor do que os demais, alcançando cerca de 70% menos do uso dos recursos de rede frente ao *Baseline*. Estes algoritmos também apresentaram alto grau de acurácia quando foram avaliados os níveis de representatividade. A Tabela 5.3 demonstra, de forma mais detalhada, os dados apresentados na Figura 5.20.

Analisando a Tabela 5.3, pode-se observar que o algoritmo DBSCAN e X-Means utilizam a menor quantidade relativa de recursos de rede e, conforme mostrado nas Figuras 5.9, 5.10 e 5.11, os mesmos mantiveram os melhores níveis de acurácia representados na identificação e classificação dos principais pontos de congestionamento em todas as densidades analisadas. Adicionalmente, os algoritmos mantiveram um comportamento constante frente ao algoritmo *Baseline*. Entre os algoritmos baseados em amostragem simples, pode-se observar que o aumento da densidade pode ter impacto significativo no consumo dos recursos, representando índices de consumo próximos a 80%. Pode-se inferir ainda, que em cenários de maior densidade veicular, tais abordagens tendem a amplificar o consumo dos recursos da rede.

De acordo com os resultados da simulação, os métodos baseados em agrupamento fornecem uma classificação assertiva de fluxo de tráfego quando analisados sob a métrica

Tabela 5.3: Tabela Comparativa da Largura de Banda e a Quantidade de Veículos.

Densidade $km^2$	100		200		300	
	Cons. Link (KB/seg)	%	Cons. Link (KB/seg)	%	Cons. Link (KB/seg)	%
Baseline	39790,97	100	43396,09	100	30596,19	100
Lim	37924,2	95,31	40869,02	94,18	32423,09	105,97
1to2	37005,38	93	40051,66	92,29	28299,04	92,49
Rand	35304,75	88,73	37666,98	86,8	28987,41	94,74
HDBSCAN	28842,3	72,48	28827,3	66,43	20223,62	66,1
OPTICS	24453,15	61,45	23436,81	54,01	17024,22	55,64
X-Means	4012,18	10,08	3874,86	8,93	4066,52	13,29
DBSCAN	4512,98	11,34	4860,47	11,2	5022,21	16,41

LOS, com menos uso de recursos de rede do que outras abordagens, como 1to2, Rand e Lim. Esse comportamento é corroborado por um trabalho anterior (PEIXOTO et al., 2020). Neste trabalho, os autores analisam a relação entre o uso da rede e o número de grupos realizados pelo DBSCAN.

Os autores constataram em seus estudos que o aumento do número de veículos e, conseqüentemente, do número de aglomerados, indica um menor número de pontos de ruído. Além disso, com o aumento do número de veículos, o número de veículos por *cluster* também aumenta, reduzindo proporcionalmente a quantidade de dados enviados por grupo. Portanto, pode-se mostrar com base nos experimentos apresentados que o número de *clusters*, densidade veicular e LOS são significativamente interdependentes.

Adicionalmente, pode-se observar ainda na Figura 5.20 uma tendência inversa dos dados resultantes do algoritmo DBSCAN frente ao algoritmo *Baseline*. A correlação fornece indício de que conforme a densidade aumenta, menos recursos de rede são requeridos. Isso pode ser explicado pelo fato de que quando há um número maior de veículos no cenário, há mais possibilidades de congestionamento e, conseqüentemente, mais redução no consumo da rede.

Observa-se, portanto, que os algoritmos Lim, 1to2 e Rand são capazes de reduzir o uso da rede em relação ao *Baseline*, mas com um custo maior do que os algoritmos baseados em agrupamento. Tanto o X-Means quanto o DBSCAN reduziram o tráfego de dados em cerca de 87% e 84,41% em relação ao *Baseline*, respectivamente. Além disso, o DBSCAN reduz 78,33% mais dados de tráfego do que o Rand, que é a melhor abordagem para redução de dados de tráfego na literatura.

Analisando essas taxas de redução apresentadas sob a perspectiva dos custos de comunicação relativos a *Operational Expenditure* (OPEX), pode-se observar os efeitos práticos dessas abordagens a partir de duas perspectivas concorrentes: Primeiro, a) armazenamento de dados, que está relacionado à quantidade de dados necessários para que um sistema de transporte inteligente possa realizar análises e identificar eventuais mudanças no comportamento do tráfego. Em segundo lugar, b) o custo de comunicação, que está relacionado ao comportamento do tráfego de dados pela rede durante o monitoramento

e gerenciamento dos espaços urbanos.

Analisando as taxas de redução apresentadas sob a perspectiva de armazenamento de dados, pode-se observar um impacto significativamente relevante. Estudos demonstram que um veículo conectado por um sistema de transporte inteligente pode exigir cerca de 1 TB de dados por dia (XU et al., 2018). Correlacionando esses valores com o preço médio do armazenamento em nuvem, que é de cerca de \$24,00/mês cada 1 TB, é possível supor que o custo máximo será de  $1 \text{ TB} * 30 \text{ dias} = 30 \text{ TB/mês}$  por veículo. Em um cenário de 70 veículos, esse valor pode chegar a 10.800TB, gerando um total anual de cerca de US\$ 756.000 apenas para armazenamento de dados. Estes números são significativamente maiores em um cenário de veículos autônomos.

Outro aspecto a ser observado consiste no aumento do número de veículos conectados e automatizados (CAV). O CAV é uma tecnologia transformadora que tem grande potencial para reduzir acidentes de trânsito, melhorar a qualidade de vida e a eficiência dos sistemas de transporte. Estima-se que quase 70% dos veículos novos em todo o mundo terão uma interface de conectividade integrada à uma rede celular que fornecerá dados veiculares de gateways, softwares ou sensores (GINO, 2006). Considerando que os dados gerados por CAVs podem ultrapassar 2 GB por segundo, esse crescimento exponencial de dados é inevitável, aumentando o custo de comunicação da rede e a complexidade computacional para orquestração de dados (por exemplo, armazenamento, processamento e análise) e extração de conhecimento. Assim, ao analisar a métrica de redução de dados proposta neste experimento frente aos valores cobrados por este serviço em 2022, pode-se notar uma redução relevante nos valores. A figura abaixo ilustra esta análise.

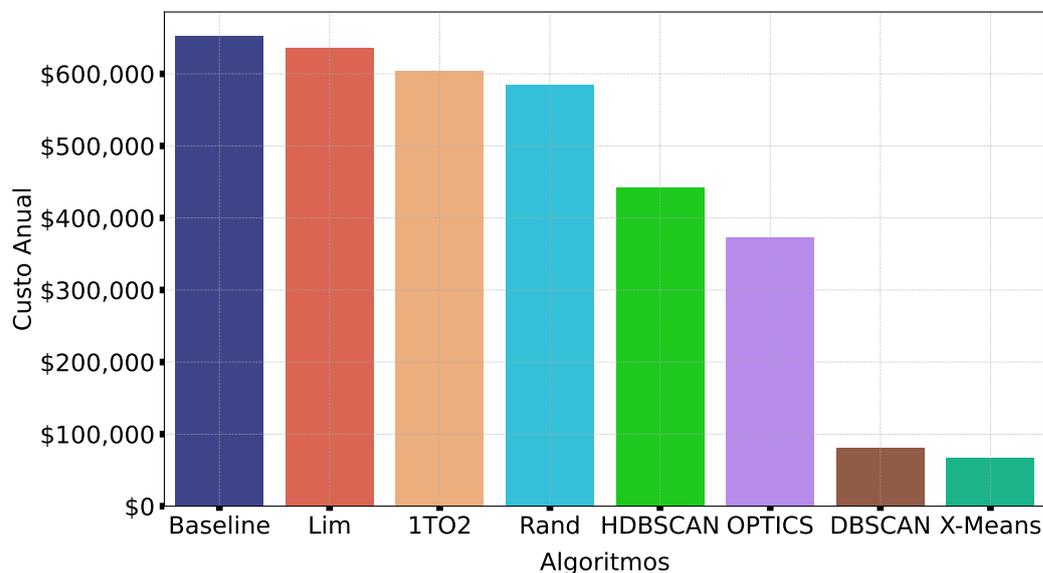


Figura 5.21: Estimativa de Custos de Armazenamento

Além disso, conforme mencionado anteriormente, existem também os custos relacionados ao tráfego de dados pela rede. Nesta proposta, enfatizam-se os custos no enlace entre a Névoa e a Nuvem. Neste contexto, temas como taxa de transição, largura de banda ou taxa de uso do *spectro* tornam-se especialmente relevantes. Cada célula 5G é

capaz de suportar pelo menos um milhão de dispositivos por Km<sup>2</sup>. Assim, a velocidade de download e upload é dividida igualmente entre todos os dispositivos conectados na mesma célula (SHAW; BELTRÁN; SOWERBY, 2017).

No contexto da computação urbana, centenas de veículos podem compartilhar este mesmo canal de comunicação e cada um deles é capaz de emitir dados a uma taxa de transição de 1 Mhz. Esses dados são recebidos pela RSU e enviados para o processamento na infraestrutura de Névoa ou Nuvem. Essa carga de trabalho aliada ao uso dos diferentes dispositivos nesta mesma rede podem levar à sobrecarga e a consequente perda de qualidade em função da desigualdade na taxa de uso do *Spectro*. Assim, estratégias de redução de dados como DBSCAN, X-Means, OPTICS ou HDBSCAN podem contribuir para mitigar esse aumento de custo associado, reduzindo em cerca de 70% o uso dos recursos endereçados à rede veicular. Essa ação pode contribuir para o aumento da qualidade geral *Quality of Service* (QOS) do serviço e gerar um melhor aproveitamento dessa infraestrutura.

### 5.6.2 Latência

Embora o consumo de rede seja um dos aspectos mais relevantes da abordagem proposta neste trabalho, outros aspectos do uso dos recursos de rede podem ser avaliados. Assim, a Figura 5.22 ilustra o perfil da latência aferida durante o tráfego de dados na rede para cada um dos algoritmos, variando a densidade veicular nas mesmas proporções supracitadas.

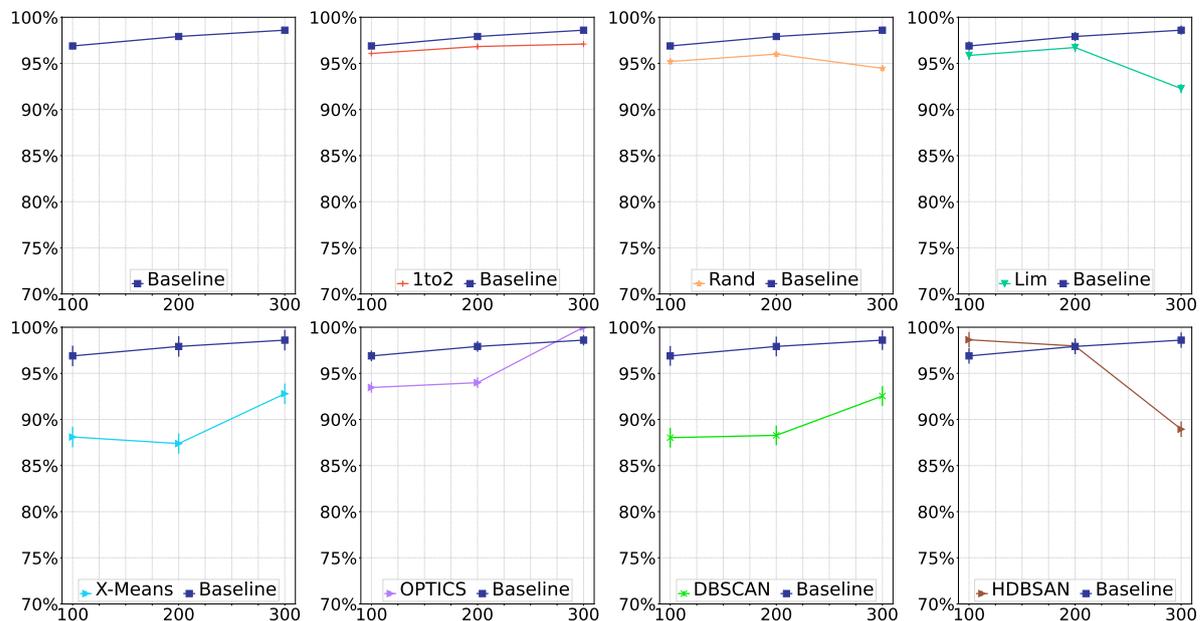


Figura 5.22: Latência 100, 200 e 300 veículos/ $km^2$ .

Analisando a Figura 5.22, pode-se observar os níveis de latência no uso da rede durante os experimentos realizados. Em redes de computadores, a latência consiste no atraso ou tempo que uma requisição leva para trafegar de um ponto para outro em uma rede interconectada. Assim, estas medições referem-se ao atraso no envio dos dados ao longo

do experimento. A fim de mensurar o impacto do aumento do fluxo de dados e das estratégias de redução na latência de rede, as densidades são apresentadas no eixo  $x$  do gráfico para cada algoritmo, separadamente.

Pode-se observar na Figura 5.22, que os algoritmos 1to2, Rand e Lim denotam uma ligeira melhora, conforme o volume de dados aumenta. No entanto, em todos os casos, os algoritmos mantêm níveis de latência próximos ao *Baseline*. No tocante aos algoritmos baseados em agrupamento, pode-se observar que os algoritmos X-Means e DBSCAN mantiveram uma queda mais acentuada no tempo de resposta do que os algoritmos OPTICS e HDBSCAN.

Adicionalmente, cabe observar que os algoritmos OPTICS, X-Means e HDBSCAN apresentaram aumento da latência, conforme a densidade veicular é ampliada. Nestes casos, o algoritmo DBSCAN manteve menor aumento proporcional do que os demais. O algoritmo HDBSCAN demonstrou uma tendência inversa, denotando que, conforme a densidade veicular aumenta, o atraso no tráfego de dados será reduzido. Experimentos com maiores variações de densidade são necessário para comprovar tal hipótese.

### 5.6.3 Pacotes Trafegados

A fim de observar outras métricas relevantes relativas ao tráfego de dados na rede, a Figura 5.23 ilustra, para este mesmo cenário, o comportamento da rede ao mensurar a taxa de envio de pacotes ao longo do processamento de dados realizado pelos algoritmos analisados neste trabalho.

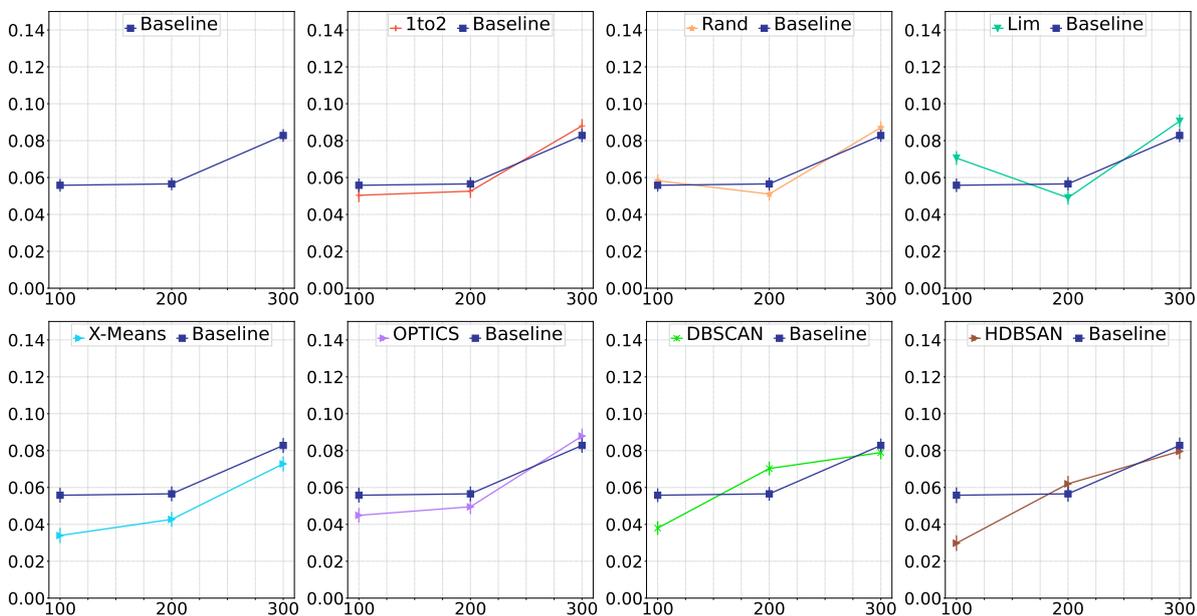


Figura 5.23: Taxa de envio de pacotes 100, 200 e 300 veículos/ $km^2$ .

Em uma rede de computadores, a informação a ser trafegada pela rede é, em geral, particionada em porções menores que são comumente chamados de pacotes. O recebimento correto dos dados enviados depende, em grande medida, da capacidade do consumidor

conseguir organizar os pacotes recebidos e reconstruir o conjunto original. Neste contexto, mitigar as ocorrências relativas à perda de pacotes é fator preponderante para garantir a qualidade do tráfego de dados em toda a rede. Diante disso, a Figura 5.23 ilustra o comportamento da rede por meio de medições baseadas na taxa de envio de pacotes ao longo do experimento, variando as densidades, conforme mencionado anteriormente.

Analisando a Figura 5.23, pode-se notar que entre os algoritmos baseados em amostragem simples como 1to2 e Rand, a taxa de envio de pacotes é similar ao algoritmo *Baseline*. Para estes algoritmos, a análise sugere um aumento sistemático em função da densidade veicular. O algoritmo Lim apresentou uma taxa de envio ligeiramente superior para uma densidade de 200 veículos. Entre os algoritmos baseados em agrupamento, observa-se que o DBSCAN demonstrou uma tendência diferente dos demais, apresentando uma redução na taxa de envio de pacotes em função do aumento da densidade. No entanto, entre as densidades de 200 e 300, o aumento foi menos significativo. O algoritmo HDBSCAN manteve um comportamento similar. As melhores taxas de envio foram observadas durante a execução dos algoritmos X-Means e OPTICS em todas as replicações.

#### 5.6.4 Throughput

Em se tratando do perfil dos algoritmos em relação à taxa de throughput, a Figura 5.24 ilustra o comportamento da rede no que se refere a tais medições variando as densidades entre 100, 200 e 300 veículos/ $km^2$  ao longo dos experimentos e replicações analisadas.

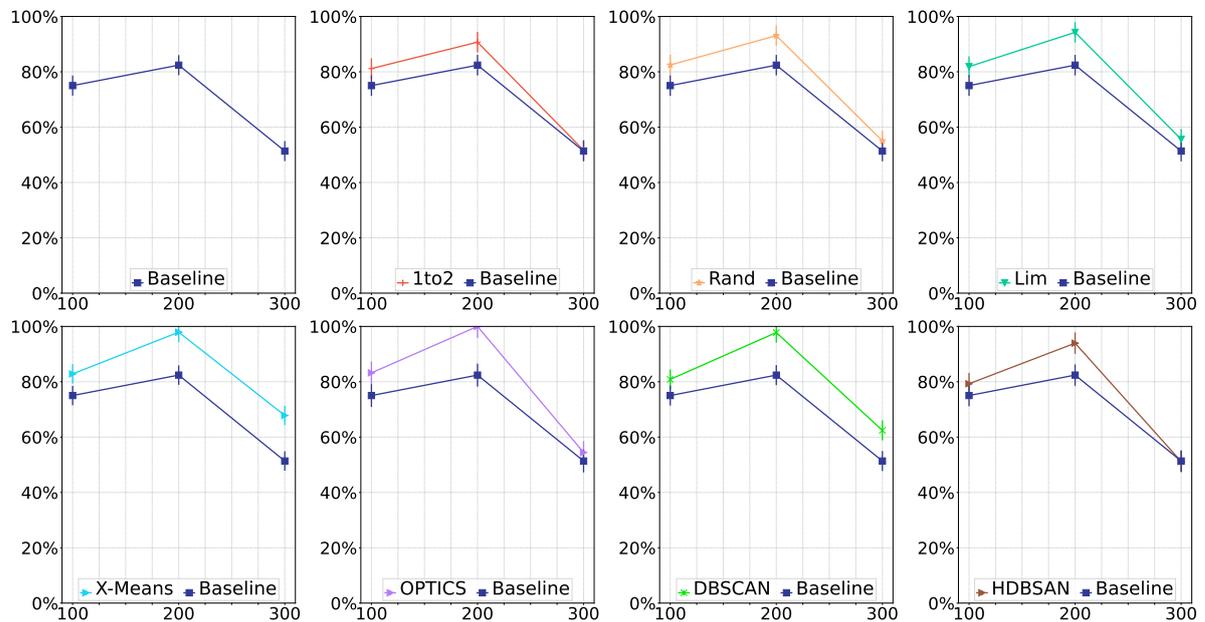


Figura 5.24: Monitoramento do Throughput para 100, 200 e 300 veículos/ $km^2$ .

Analisando a Figura 5.24, pode-se observar que a taxa de *throughput* mostrou-se superior em todos os algoritmos. Em redes de computadores, o *throughput* consiste na aferição da quantidade de dados transferidos com êxito durante um determinado intervalo

de tempo. Assim, embora exista uma forte correlação entre o *throughput* e a largura de banda, as métricas em si são diferentes, uma vez que o envio dos dados por uma certa rede de computadores está sujeito a perdas de pacotes e atrasos durante o envio e recebimento de dados. As medições de *throughput* tratam dessa efetiva capacidade de comunicação.

Cabe observar ainda, que o *throughput* manteve-se estável entre as densidades de 100 e 200 veículos. No entanto, quando essa densidade foi ampliada para 300 veículos, ocorreu uma redução em torno de 30% na taxa de transferência efetiva. Este fenômeno pode ser explicado pelo fato de que, quando a densidade é ampliada para 300 veículos, ocorre uma propagação de congestionamentos em diferentes rodovias, aumentando o tráfego de dados no cenário. Esse aumento reduz a capacidade da rede em lidar com o volume de dados, reduzindo assim a taxa de *throughput*.

Cabe citar ainda, que entre os algoritmos baseados em amostragem simples, os algoritmos 1to2, Rand e Lim registraram taxas similares ao *Baseline*. Nestes casos, quando a densidade evolui para 300 veículos, os valores tornam-se, em alguns casos, sobrepostos. Para os algoritmos baseados em agrupamento, observa-se que o X-Means e o DBSCAN registraram um desempenho ligeiramente superior do que os demais. Nestes casos, mesmo quando a densidade evolui para 300 veículos, estes algoritmos registram taxas que chegam a ser 20% melhores do que todos os algoritmos analisados.

## 5.7 VOLUME DE DADOS GERADOS

Com o objetivo de avaliar o desempenho de cada um dos algoritmos no que se refere a redução efetiva de dados, a Figura 5.25 ilustra, em escala percentual, a taxa de redução dos dados para cada um dos algoritmos utilizando replicações em um intervalo de confiança de 95%.

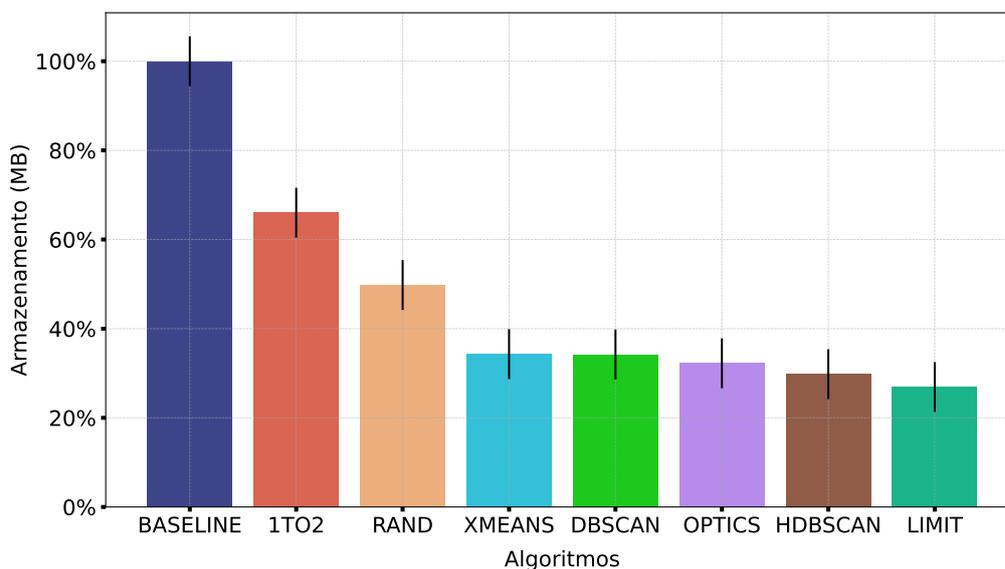


Figura 5.25: Média da Redução Durante os Experimentos

Analisando a Figura 5.25, pode-se perceber que, exceto pelo algoritmo Lim, os al-

goritmos baseados em amostragem simples demonstraram redução efetiva menor do que os algoritmos baseados em agrupamento. Analisando o algoritmo 1to2, observa-se que a redução efetiva chegou a 66%, denotando que apenas a utilização dessa estratégia reduziria quase à metade a quantidade de dados processados, garantindo ainda níveis de acurácia eficientes. Para o algoritmo Rand, esta redução torna-se ainda maior, alcançando algo próximo a 50% dos dados trafegados pelo algoritmo *Baseline* e mantendo níveis de acurácia satisfatórios.

Embora o algoritmo Lim demonstre agressiva taxa de redução efetiva frente aos demais algoritmos baseados em amostragem, sua capacidade de representar o conjunto de dados original mostrou-se limitada, tanto na análise da classificação por nível de serviço LOS, quando na representatividade ilustrada na Figura 5.12 por meio do cálculo do erro quadrático médio, não demonstrando ser uma opção viável para identificação de congestionamento em um sistema de transporte inteligente. No que se refere aos algoritmos baseados em agrupamento, pode-se perceber níveis similares de redução nos conjunto de dados resultantes. Neste contexto, os algoritmos OPTICS e HDBSCAN demonstraram níveis de redução ligeiramente maiores do que os demais, no entanto, quando estes dados são analisados frente ao perfil de uso da rede Figura 5.20, observa-se que os algoritmos DBSCAN e X-Means mantiveram taxas de utilização de banda significativamente menores em função das suas estratégias de processamento de dados e seleção de elementos. Quando confrontados com as medidas de acurácia e classificação por níveis de serviço LOS, estes resultados também demonstram melhor desempenho nestes algoritmos, com o DBSCAN mantendo uma ligeira superioridade frente ao algoritmo X-Means.

Outro aspecto relevante a ser observado consiste na análise do tamanho dos conjuntos de dados resultantes frente ao aumento da densidade veicular. Esta relação é ilustrada na Figura 5.26

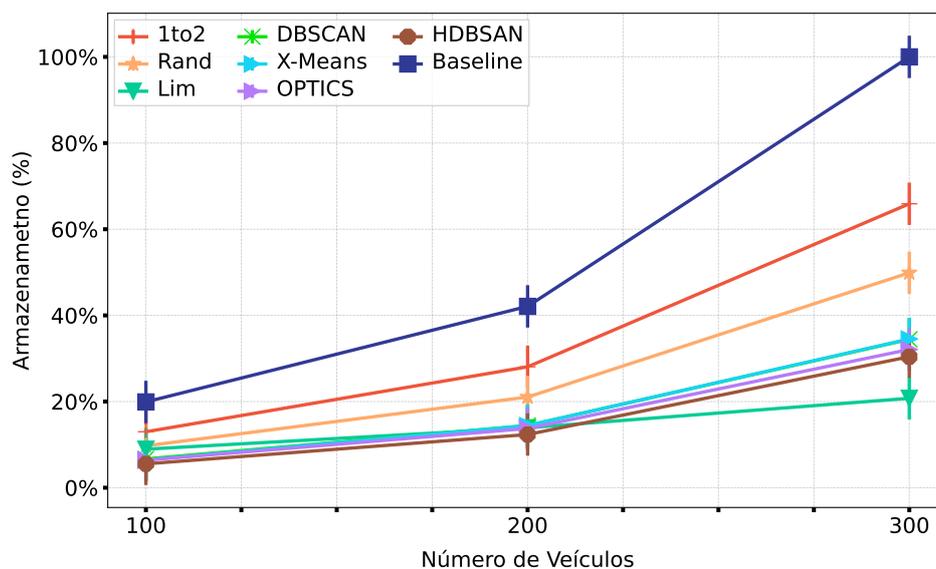


Figura 5.26: Relação entre o tamanho do arquivo e a densidade veicular

Analisando a Figura 5.26, pode-se observar que em todos os casos ocorreu um aumento

significativo do conjunto de dados resultante em função da densidade veicular. Entre os algoritmos de amostragem simples, o processo de redução torna-se mais eficiente a medida que a densidade veicular aumenta. Analisando o algoritmo 1to2, observa-se que a sua taxa de redução é menor que 10% em um cenário com densidade de 100 veículos, quando essa densidade é aumentada para 300 veículos, essa taxa sobe para algo em torno de 30%. Este mesmo fenômeno corre com o algoritmo Rand.

Quando a densidade veicular variar entre 200 e 300 veículos/ $km^2$ , ocorre um aumento mais significativo do volume de dados, este aumento se dá em função da formação de congestionamentos que mantêm os veículos por mais tempo dentro do cenário de simulação. Nestes casos, pode-se observar que o algoritmo 1to2 demonstra uma tendência de crescimento mais acentuada, denotando que, entre os algoritmos baseados em amostragem simples, o Rand é relativamente superior.

Entre os algoritmos baseados em agrupamento, pode-se observar que o aumento da densidade tem menor influência na taxa de redução alcançada, sendo esta sensivelmente mais baixa do que as taxas observadas nos algoritmos baseados em amostragem simples. Adicionalmente, ao analisar apenas a variação da densidade entre 200 e 300 veículos/ $km^2$ , pode-se observar que os algoritmos OPTICS, HDBSCAN, X-Means e DBSCAN mantiveram desempenho similar, mantendo uma tendência de crescimento mais atenuada do que os demais algoritmos.

A seção seguinte apresenta o desempenho das estratégias de roteamento utilizado as diferentes abordagens, completando o ciclo de vida do *Framework* DaRkaM proposto neste trabalho.

## 5.8 PROCESSO DE RE-ROTEAMENTO E TOMADA DE DECISÃO

Este trabalho tem por objetivo propor um *framework* capaz de promover a redução de dados na névoa, atenuando o uso dos recursos de rede ao passo que reduz a latência e os custos de comunicação. Entretanto, conforme mencionado anteriormente, a redução de dados deve atuar em duas perspectivas concomitantes, de um lado, a efetiva redução do uso dos recursos de rede e de outro, a seleção assertiva de um conjunto de elementos capazes de garantir altos níveis de representatividade frente a um conjunto de dados original. Ao avaliar estas duas perspectivas face a um sistema de transporte inteligente, pode-se observar a necessidade de um terceiro e determinante fator para avaliar a qualidade das estratégias de redução implementadas nesse trabalho, o uso dos dados processados na tomada de decisão.

Um sistema de transporte inteligente tem por objetivo maximizar a eficiência do fluxo de tráfego por meio da seleção, análise e eventuais intervenções no cenário urbano. A tomada de decisão ocorre por meio da identificação de certas características contidas nos conjuntos de dados coletados que permitem a avaliação e elaboração de alternativas para maximizar a eficiência do tráfego. No centro das estratégias utilizadas para maximizar a eficiência do fluxo de tráfego das rodovias persiste o conceito de re-roteamento.

Conforme descrito na Figura 4.1, o *framework* DaRkaM foi elaborado com base em uma estrutura multicamada na qual os dados do ambiente de mobilidade são capturados por dispositivos posicionados na borda da rodovia chamados de *Road Side Unit*(RSU) e

então enviados para a camada de névoa. Uma vez na névoa, estes dados são processados e as estratégias de redução são aplicadas com o fim de reduzir o custo no enlace entre a névoa e a nuvem. Uma vez na nuvem, estes dados são utilizados para a tomada de decisão. Na nuvem, um módulo de re-roteamento é capaz de gerar e distribuir novas rotas com base no cenário atual, todo esse processo ocorre em um fluxo de dados contínuo durante toda a simulação.

A fim de analisar a eficiência do *framework* DaRkaM na geração de dados para a tomada de decisão, a Figura 5.27 ilustra o comportamento do ambiente de mobilidade quando o serviço de distribuição de rotas é acionado. O cálculo das possíveis rotas foi realizado com base no algoritmo *k shortest path*. Este algoritmo garante uma análise do caminho mais curto por meio de um gráfico ponderado capaz de fornecer  $K$  rotas priorizadas de um ponto origem a um destino.

A partir dessa abordagem, tornou-se possível definir o conjunto de rotas que são distribuídas no ambiente de mobilidade com base em um sistema de peso dinâmico atribuído à rodovia e faixa. Os pesos são atribuídos com base no nível de congestionamento detectado. O padrão de classificação por nível de serviço LOS foi utilizado para denotar a qualidade do fluxo de tráfego durante todo contexto da simulação.

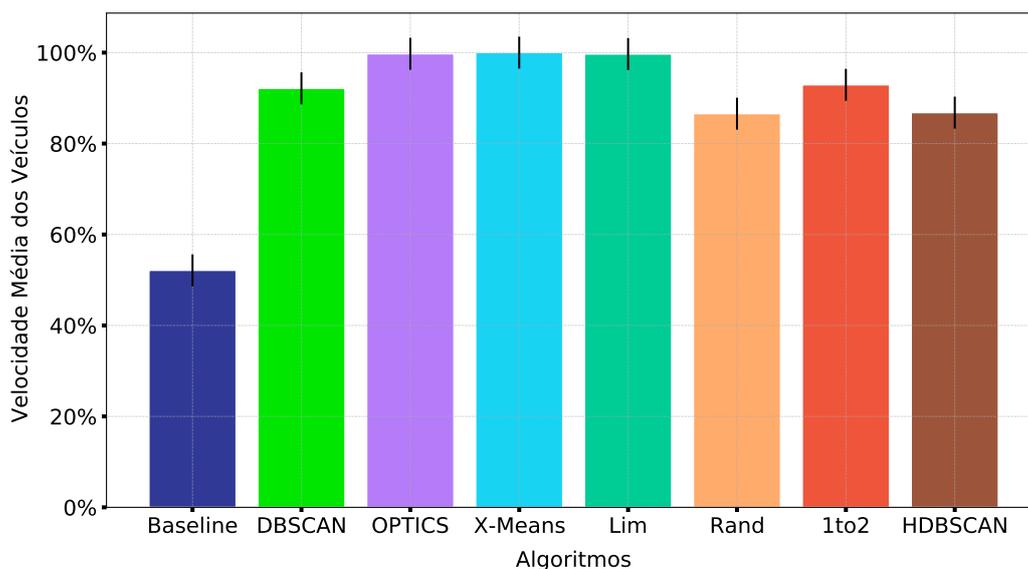


Figura 5.27: Velocidade média das rodovias durante a simulação

A Figura 5.27 ilustra a variação da velocidade da via ao longo dos experimentos realizados, tende-se a acreditar que quanto maior a velocidade média dos veículos, melhor o fluxo de tráfego nas rodovias analisadas. Os valores foram obtidos por meio da execução das simulações e replicações, com o fim de identificar a variabilidade dos dados coletados.

Pode-se observar que a velocidade média dos veículos aumentou significativamente em todos os experimentos quando comparada ao algoritmo *Baseline*. Entre os algoritmos baseados em agrupamento, observa-se que os algoritmos X-Means e OPTICS, alcançaram os melhores resultados, apresentando um aumento na velocidade média em torno de 50%

em relação ao algoritmo *Baseline*. O algoritmo HDBSCAN obteve um desempenho ligeiramente inferior. Analogamente, entre os algoritmos baseados em amostragem simples, destaca-se o algoritmo Limite, seguido do algoritmo 1TO2. O algoritmo Rand registrou um aumento significativo em relação ao algoritmo *Baseline*, no entanto, em torno de 10% inferior, quando comparado com os demais algoritmos baseados em amostragem simples. Corroborando com essa análise, a Figura 5.28 ilustra o tempo de viagem obtido pelo roteamento em cada um dos algoritmos analisados.

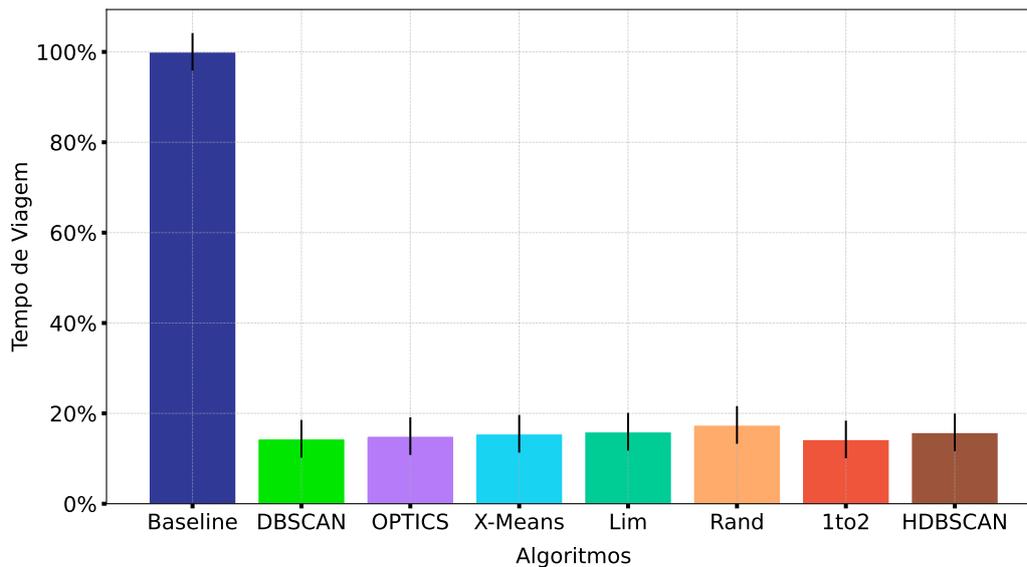


Figura 5.28: Tempo de viagem

Ao analisar o tempo que um veículo leva da sua origem ao seu destino, pode-se observar que todos os algoritmos, nos quais foram empregadas as técnicas de roteamento, mantiveram um desempenho significativamente superior, registrando reduções na ordem de 70%, frente ao algoritmo *Baseline*. Nesse contexto, o algoritmo DBSCAN apresentou um desempenho ligeiramente superior entre os algoritmos de agrupamento, seguido dos algoritmos OPTICS, X-M-Means e HDBSCAN, respectivamente. Entre os algoritmos baseados em amostragem simples, o algoritmo 1to2 registrou o melhor desempenho entre todos os algoritmos analisados.

Assumindo que um dos principais objetivos deste trabalho consiste em estabelecer um modelo de redução de dados capaz de reduzir os custos de comunicação, sem impactar nos níveis de detecção de congestionamento e garantindo a tomada de decisão, é correto afirmar que as estratégias de redução de dados utilizando os algoritmos baseados em agrupamento, em especial os algoritmos DBSCAN e X-Means, no contexto de fluxo de dados contínuo, mostrou-se realmente promissora.

## 5.9 DISCUSSÃO E CONCLUSÃO

Os resultados apresentados ao longo deste capítulo demonstram os diferentes comportamentos dos algoritmos frente às diferentes métricas aferidas durante os experimentos.

Tabela 5.4: Análise comparativa dos resultados por algoritmos e métricas aplicadas

	Amostragem			Agrupamento			
	1TO2	Random	Limite	DBSCAN	XMeans	OPTICS	HDBSCAN
Uso da Rede	80,925	78,316	85,427	11,057	9,181	49,861	59,831
Arm. de Dados	66,018	49,789	26,919	34,291	34,211	32,240	29,807
Erro Médio (RMSE)	17,885	24,347	100,00	34,107	34,553	35,297	42,342
Latência	96,665	95,222	94,950	89,612	89,422	95,817	95,181
Throughput	74,515	76,863	77,251	80,356	82,859	79,217	74,801
Envio de Pacotes	90,886	93,470	100,00	89,074	71,136	86,764	81,575
Velocidade Média	92,917	100,00	82,508	86,990	78,891	93,686	91,137
Tempo de Viagem	57,108	49,927	63,026	61,966	68,198	56,922	59,134
Custo (OPEX)	80,528	77,889	84,800	10,837	9,038	49,810	59,026
Coef. Silhueta	-	-	-	0,229	-0,102	0,573	0,568
Num. de Clusteres	-	-	-	0,528	0,191	100,00	55,197

Estes estudos contribuíram para estabelecer, entre outros aspectos, uma análise comparativa que denota, de forma geral, o desempenho de cada algoritmo frente aos demais. Entretanto, cabe observar que o desempenho de um certo algoritmo deve considerar não só o conjunto de métricas aferidas, mas principalmente, os requisitos, recursos e ambiente sob o qual uma certa abordagem será aplicada. A simples análise comparativa das métricas aferidas pode não ser suficiente para fornecer essa perspectiva particular acerca do desempenho das abordagens em um certo caso de uso. Neste contexto, pode-se concluir, por exemplo, que uma aplicação poderá priorizar a acurácia, em detrimento a redução de dados, enquanto em outra, deseja-se priorizar a baixa latência e assim sucessivamente. Desta forma, a escolha do algoritmo mais adequado passa inevitavelmente pelo cumprimento destes requisitos essenciais de cada aplicação. A fim de estabelecer uma análise comparativa entre as métricas e algoritmos analisados ao longo deste capítulo, a Tabela 5.4 ilustra cada uma das métricas frente aos algoritmos implementados no *framework* DaRkaM.

Analisando a Tabela 5.4, pode-se observar que cada algoritmo está relacionado com todas as métricas aferidas durante os experimentos realizados ao longo deste capítulo. Para facilitar a análise dos dados, os resultados foram convertidos em unidades percentuais, e seus valores foram estabelecidos com base nos resultados comparativos frente ao algoritmo *Baseline*. Além disso, os valores máximos foram extraídos das métricas, a fim de estabelecer um ponto percentual máximo para cada algoritmo em relação aos demais, estes valores foram representados na tabela por 100%. Pode-se observar ainda, que os resultados foram divididos em dois grupos, os algoritmos cujo processamento é baseado

em um sistema de amostragem simples e os algoritmos baseados em agrupamento. Conforme já mencionado, para cada um deles foram calculados os percentuais com base nos resultados obtidos a partir dos experimentos realizados.

Entre os algoritmos baseados em amostragem simples, pode-se observar relativa variação entre algumas métricas. Analisando os valores do algoritmo **1TO2**, pode-se observar altos níveis de acurácia, alcançando o menor percentual para a *Raiz do Erro Quadrático Médio* RMSE entre todos os demais algoritmos. Esta representatividade é seguida pelo algoritmo **Random** com 24.3% e subsequentemente, pelo algoritmo de agrupamento **DBSCAN**, com 34.1%, uma diferença em torno de 16% para o algoritmo de menor RMSE. No entanto, ao analisar estes resultados frente ao consumo dos recursos de rede, pode-se observar que o algoritmo **1TO2** mantém taxas significativamente maiores do que as observadas no algoritmo **DBSCAN**, sendo este capaz de alcançar uma taxa de redução próxima a 70% frente ao algoritmo **1TO2**. A melhor relação entre a taxa de uso da rede e a acurácia (RMSE), pode ser observada no algoritmo **X-Means**, uma vez que o mesmo apresenta um consumo de rede na ordem de 9,1% em relação ao **Baseline** e ao mesmo tempo, mantém altos níveis de acurácia, desempenhando uma diferença de 16% para o algoritmo **1TO2**. Esses resultados são seguidos de perto pelo algoritmo **DBSCAN**.

Ao analisar o armazenamento de dados, pode-se observar que o algoritmo **HDBSCAN** manteve uma posição de destaque. Embora seu consumo dos recursos de rede seja superior aos algoritmos **DBSCAN** e **X-Means**, os conjuntos de dados resultantes dessa abordagem alcançaram os melhores resultados, mantendo níveis de acurácia igualmente representativos. O algoritmo Limite manteve uma taxa de armazenamento similar, porém, obteve taxa de erro inferior entre os algoritmos analisados e o maior consumo dos recursos de rede.

Ao analisar os recursos relativos ao desempenho da rede, pode-se observar que o algoritmo **X-Means** apresentou a maior taxa de *throughput* entre os algoritmos analisados, seguido pelo algoritmo **DBSCAN**, com 80.3% e pelo algoritmo **OPTICS**, com cerca de 79,2%. Entre os algoritmos baseados em amostragem simples, o **Random** manteve a maior taxa de *throughput* e concomitantemente a menor taxa de latência entre os algoritmos analisados. Apesar destes valores terem sido superados pelo algoritmo limite nos experimentos avaliados, seus níveis de acurácia e uso da rede inferem baixa aplicabilidade dessa abordagem no campo dos sistemas de transporte inteligente. Análogamente, a maior taxa de envio de pacotes foi alcançada pelo algoritmo **Limite**, seguido pelo algoritmo **Random** e **1TO2**. Entre os algoritmos de agrupamento, a melhor taxa foi observada no algoritmo **OPTICS** alcançando 86.7%, uma diferença de 6,7% para o algoritmo **Random** que obteve o melhor desempenho nessa métrica.

Outro importante indicador analisado durante os experimentos tem por fim mensurar a atuação do *framework* DaRkaM no processo de tomada de decisão, avaliando sua contribuição na distribuição das rotas de forma a promover maior fluidez geral do trânsito. Neste ponto, dois indicadores foram mapeados, são eles a *velocidade média* da rodovia e o *tempo geral das viagens* realizadas ao longo das simulações. Analisando a velocidade média, pode-se observar que o algoritmo **Random** obteve o melhor desempenho no que se refere ao aumento percentual da velocidade geral dos veículos nas rodovias, tendo como

consequência, o menor tempo médio de viagens registrado durante os experimentos. No entanto, cabe destacar que estes resultados foram seguidos pelo algoritmo **OPTICS** que manteve uma diferença de 6.9% na velocidade média e uma diferença de 7% no tempo de viagem, tudo isso alcançando resultados satisfatórios na acurácia (RSME), com significativa redução do uso de recurso de rede e do armazenamento de dados.

Os recursos utilizados para processamento dos dados em fluxo contínuo, preconizado pelo *framework* DaRkaM, podem implicar na redução de custos de armazenamento e comunicação, estes custos são, em geral, descritos como custos operacionais ou OPEX. Analisando a métrica de custo entre os algoritmos, pode-se observar que o menor custo operacional foi alcançado pelo algoritmo **X-Means**, seguido do algoritmo **DBSCAN** com uma diferença de apenas 1,8% entre os dois. Por outro lado, o maior custo operacional foi representado pelo uso do algoritmo **Limite**, seguido do algoritmo **1TO2**. Pode-se observar, portanto, que os algoritmos de agrupamento mantiveram desempenho significativamente superior frente aos algoritmos baseados em amostragem simples, especialmente, os algoritmos **DBSCAN** e **X-Means**. Estes algoritmos apresentaram taxas de redução de custo operacional que chegam a -75% do algoritmo de maior custo operacional e de -90% em relação ao algoritmo **Baseline**.

Por fim, foram analisados ainda o comportamento dos agrupamentos na formação de cluster a partir da aferição do coeficiente da silhueta e do número de clusters formados durante o processamento. O cálculo do coeficiente de silhueta consiste em quantificar tanto a coesão quanto a separação entre grupos de instâncias selecionadas, seus valores são distribuídos entre  $[-1...1]$ , onde os menores valores tendem a representar maiores níveis de coesão. Nessa perspectiva, pode-se observar que o algoritmo **X-Means** apresentou os melhores índices de coesão, seguido pelo algoritmo **DBSCAN** com 9,03% e 10,8%, respectivamente. Os algoritmos **OPTICS** e **HDBSCAN** mantiveram coeficientes similares entre si, no entanto, pode-se perceber que o processamento realizado pelo algoritmo **OPTICS** resultou em uma quantidade significativamente maior de clusters formados do que o processamento realizado pelo **HDBSCAN**, que frente ao primeiro, realizou o processamento com 55% menos clusters identificados.

Essa observação comparativa tem por objetivo contribuir para uma visão geral acerca das características dos algoritmos analisados. Este estudo pode contribuir para o desenvolvimento de aplicação não só no contexto dos sistemas de transporte inteligente, mas também, em aplicações gerais, que aspiram a realização de processamento na borda da rede como forma de atuar no universo da computação distribuída. O conjunto de indicadores apresentados podem contribuir para direcionar a escolha do algoritmo frente a um certo contexto de aplicação e essa análise conjunta maximizar os benefícios relativos a abordagem selecionada. A próxima seção aborda as considerações finais desta pesquisa.

## 5.10 CONSIDERAÇÕES FINAIS

Os resultados apresentados nesta seção demonstraram-se promissores, no que refere ao uso de algoritmos de agrupamento nos processos de seleção, detecção e tomada de decisão em um sistema de transporte inteligente. Os experimentos apresentados nessa seção consideram a implementação de oito algoritmos, dos quais quatro são baseados em amostragem

simples e os demais, baseados em agrupamento.

Ao analisar os resultados, pode-se observar que os algoritmos baseados em amostragem simples mantiveram bons níveis de representatividade, no entanto, seu desempenho em termos de redução de dados é significativamente inferior às taxas observadas nas abordagens baseadas em agrupamento.

Os experimentos se iniciaram com a escolha de um recorte de mapa representando uma área adjacente a *Cologne*, cidade da Alemanha. Sob este cenário, foram estruturados os experimentos envolvendo o *framework* DaRkaM, proposto nesse trabalho. Nesse contexto, cada experimento foi dividido em três densidades distintas, considerando 100, 200 e 300 veículos. Para cada um destes fatores, foram definidas 10 replicações, a fim de garantir um intervalo de confiança para as métricas analisadas.

A análise das variações de velocidade demonstrou que os algoritmos baseados em agrupamento mantiveram níveis de representatividade significativos, este fenômeno também pode ser observado na análise de classificação por níveis de serviço LOS. Observou-se ainda que conforme a densidade aumenta, os níveis de acurácia dos algoritmos de agrupamento também aumentaram.

Foram ainda analisados os valores com base no erro quadrático médio (RMSE), nesse contexto, os algoritmos baseados em amostragem simples mantiveram um desempenho superior, porém, estes são seguidos de perto pelos algoritmos baseados em agrupamento, mantendo em média uma diferença inferior a 0.05 na margem de erro identificada entre as abordagens. Analogamente, ao analisar estas medidas frente ao consumo de rede, pode-se observar ganhos na ordem de 80% quando comparado aos algoritmos baseados em amostragem simples. O algoritmo DBSCAN demonstrou um desempenho ligeiramente melhor entre as abordagens analisadas.

Por fim, estes dados foram utilizados para gerar intervenções no fluxo de tráfego em um contexto de fluxo de dados contínuo. Os resultados demonstraram que todos os algoritmos foram capazes de promover ganhos relacionados ao tempo de viagem, promovendo, adicionalmente, um aumento na velocidade geral das rodovias.

## CONCLUSÕES

O congestionamento do tráfego é um dos maiores problemas para os cidadãos que vivem nas grandes cidades em todo o mundo, levando a problemas econômicos e sociais nos centros urbanos. Frente a este desafio, os sistemas de transporte inteligente utilizam uma grande quantidade de dados para medir a origem e a gravidade do fluxo de tráfego, aumentando os custos da rede e sobrecarregando a infraestrutura de link de rede existente, especialmente da borda da rede para a nuvem. Para superar este problema, a presente proposta traz em seu escopo o *framework* DaRkaM, um *framework* de redução de dados na névoa criado para abordar o problema de redução do custo de comunicação em um sistema de gerenciamento de tráfego inteligente.

Vale destacar que os algoritmos baseados em amostragem simples que se utilizam de uma abordagem tradicional, são capazes de atuar no processo de redução de dados, entretanto, quando comparados com as abordagens baseadas em agrupamento, observa-se um desempenho significativamente superior no que se refere a redução de dados. Em uma abordagem baseada em agrupamento, quanto maior a densidade, menos informações proporcionais são transmitidas da névoa para a nuvem devido ao número crescente de veículos por grupo. Embora os algoritmos que utilizaram uma abordagem baseada em agrupamento mantivessem um desempenho equilibrado entre os experimentos realizados, foi possível observar que o algoritmo DBSCAN manteve níveis de redução e representatividade significativos, mostrando-se promissor quando aplicado a um contexto de fluxo de dados contínuo.

Espera-se que este trabalho contribua para a disseminação do uso do paradigma da computação em névoa no contexto da computação urbana, em especial no que se refere a Sistemas de Transporte Inteligente. O *framework* DaRkaM, proposto neste trabalho, combina diferentes abordagens para compor um modelo arquitetural capaz de atuar em diferentes cenários. Embora neste trabalho os ambientes urbanos sejam essencialmente simulados, toda a estrutura deste *framework* foi construída de forma a garantir um certo desacoplamento entre as camadas, permitindo que outros cenários urbanos sejam conectados a esta arquitetura. Esta ação visa não só garantir a independência do *framework*,

mas também, abrir espaço para novos experimentos em ambientes simulados ou mesmo em ambientes reais.

O principal objetivo dessa pesquisa foi construir um *framework* de redução de dados baseado na computação em névoa atuando em um contexto de fluxo de dados contínuo capaz de realizar o monitoramento, detecção de congestionamento e intervenção em um certo cenário urbano. Para isso, foi utilizado um conjunto de 8 algoritmos que se divide entre a) algoritmos baseados em amostragem simples, são eles Baseline, 1TO2, Rand e Lim e b) algoritmos baseados em agrupamento, dos quais foram utilizadas as abordagens XMEANS, OPTICS, DBSCAN e HDBSCAN. Para alcançar esse objetivo geral, os seguintes objetivos específicos precisaram ser atingidos:

- Projetar um ambiente de simulação capaz de reproduzir os cenários urbanos;
- Estruturar um padrão de conectividade para recebimento de dados em fluxo contínuo;
- Implementar as estratégias de redução de dados na névoa utilizando o conceito de janelas temporais;
- Implementar os modelos de classificação do fluxo de tráfego na névoa;
- Definir um padrão de troca de informações para enviar sugestões de roteamento para o ambiente de mobilidade;
- Validar a eficiência da abordagem proposta através de um planejamento de experimento utilizando replicações para indicar a influência dos algoritmos na acurácia e tomada de decisão em ambiente de simulação de tráfego;
- Submeter, periodicamente, essa pesquisa à avaliação pela comunidade acadêmica através da confecção e submissão de artigos científicos.

## 6.1 TRABALHOS FUTUROS

A presente proposta de pesquisa aborda a construção de um *framework* de redução de dados baseado na computação em névoa capaz de realizar o monitoramento, detecção de congestionamento e intervenção em um certo cenário urbano. Para isso foi utilizado um conjunto de 8 algoritmos que se divide entre a) algoritmos baseados em amostragem simples, são eles Baseline, 1TO2, Rand e Lim e b) algoritmos baseados em agrupamento, dos quais foram utilizadas as abordagens XMEANS, OPTICS, DBSCAN e HDBSCAN.

Sugere-se como um trabalho futuro a implementação de mecanismos capazes de minimizar os efeitos dos algoritmos de agrupamento na acurácia obtida pela classificação de nível de serviço (LOS) quando há menor densidade nas rodovias. Quanto ao uso de algoritmos baseados em densidade, sugere-se avaliar o impacto do raio de vizinhança nestas abordagens e como ele influencia nos níveis de acurácia por meio da aplicação da classificação do nível de serviço LOS. Estudos demonstraram que quando o valor  $\varepsilon$  escolhido é muito pequeno, uma grande parte dos dados de tráfego não será agrupada. Por outro lado, se o valor  $\varepsilon$  for muito alto, os *clusters* se fundem perdendo a qualidade em termos

de representatividade. Uma possível solução para este desafio pode ser calcular o valor de  $\varepsilon$  por meio de um grafo capaz de analisar a distância de  $K$  elementos, preferencialmente começando por pequenos valores  $\varepsilon$ .

Outro aspecto relevante consiste no fato de que a presente proposta de pesquisa limitou o escopo das análises no comportamento dos algoritmos de agrupamento e sua eficiência na redução de dados e capacidade de manter subconjuntos representativos frente a um conjunto de dados original. Uma importante extensão dessa pesquisa seria analisar o comportamento dos algoritmos de predição em um contexto de fluxo de dados contínuo para a identificação de congestionamentos, mensurando tanto o seu desempenho na redução de dados, quando a sua capacidade de extrair subconjuntos representativos frente a um conjunto de dados original.

Por fim, sugere-se ainda a adaptação dos conceitos apresentados nesta proposta para o universo dos veículos autônomos. Embora o *framework* apresentado nessa proposta de pesquisa seja estruturado em camadas, facilitando a customização para outros cenários, os mecanismos de comunicação irão requerer ajustes e adaptações para a sua atuação no contexto dos carros autônomos.

## 6.2 PUBLICAÇÕES

Com base no desenvolvimento do projeto de pesquisa e resultados alcançados, fez-se necessário submeter os mesmos à apreciação da academia por meio da submissão de alguns artigos abordando diferentes aspectos da presente proposta. Entre os artigos aceitos para publicação destacam-se os *journals* submetidos à *Ad Hoc Networks*, *Vehicular Communications* e outra publicação no *Theory of Natural Computing* (TCS-C), todos classificados como **qualis A1**, além da conferência *Vehicular Technology Conference* (VTC) também com **qualis A1**. As publicações e seus respectivos status constam na lista abaixo.

- Mota E, Coimbra D, Peixoto M, **Cartola FC Data Analysis: Uma Ferramenta para Simulação, Análise e Visualização de Dados para o Fantasy Game Cartola-FC**, ANAIS DO XIV SIMPÓSIO BRASILEIRO DE SISTEMAS DE INFORMAÇÃO, 2018 (Aceito)
- Coimbra D, Martins R, Mota E, Tiburtino T, Diamantino P, Peixoto M **Analyzing the Quality of Local and Global Multidimensional Projections Using Performance Evaluation Planning**, Journal Theoretical Computer Science - C (TCS-C Theory of Natural Computing), 2019 (Aceito)
- Peixoto M, Mota E, Maia A, Santos M, Junior W, Villas L **Exploiting Fog Computing with an Adapted DBSCAN for Traffic Congestion Detection System**, Vehicular Technology Conference: VTC2020-Fall, 2020 (Aceito)
- Peixoto M; Maia A; Mota E; Rangel E; Costa D; Turgut D; Villas L **A Traffic Data Clustering Framework based on Fog Computing for VANETs**, Vehicular Communications, 2021 (Aceito)

- Peixoto M; Mota E; Maia A; Lobato W; Salahuddin M; Boutaba R; Villas L **Fog-Jam: A Fog Service for Detecting Traffic Congestion in a Continuous Data Stream VANET**, Ad Hoc Networks, 2022 (Aceito)

## REFERÊNCIAS BIBLIOGRÁFICAS

- AADIL, F. et al. Caconet: Ant colony optimization (aco) based clustering algorithm for vanet. *PloS one*, Public Library of Science San Francisco, CA USA, v. 11, n. 5, 2016.
- AAZAM, M.; HUH, E. Fog computing: The cloud-iotioe middleware paradigm. *IEEE Potentials*, v. 35, n. 3, p. 40–44, May 2016. ISSN 0278-6648.
- ABRAHAM, S.; WELLER, N. Next generation traffic event detection: A framework based on internet of vehicles (ioV). In: *Proceedings of the Second International Conference on Internet of Things, Data and Cloud Computing*. New York, NY, USA: Association for Computing Machinery, 2017. (ICC '17). ISBN 9781450347747.
- AHMADZADEGAN, M. H. et al. Implementation and evaluation of the impact of traffic congestion on the detection of the missing packets in vanet. In: *2019 Third International conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC)*. [S.l.: s.n.], 2019. p. 169–172.
- AKABANE, A. T. et al. Towards a distributed and infrastructure-less vehicular traffic management system. *Computer Communications*, v. 151, p. 306 – 319, 2020. ISSN 0140-3664.
- AL-SULTAN, S. et al. A comprehensive survey on vehicular ad hoc network. *Journal of Network and Computer Applications*, v. 37, p. 380 – 392, 2014. ISSN 1084-8045. Disponível em: <http://www.sciencedirect.com/science/article/pii/S108480451300074X>.
- ALI, T.; ASGHAR, S.; SAJID, N. A. Critical analysis of dbSCAN variations. In: *IEEE. 2010 International Conference on Information and Emerging Technologies*. [S.l.], 2010. p. 1–6.
- ALKHEDER, S.; ALOMAIR, A. Urban traffic prediction using metrological data with fuzzy logic, long short-term memory (LSTM), and decision trees (DTs). *Natural Hazards*, Springer Science and Business Media LLC, v. 111, n. 2, p. 1685–1719, nov. 2021. Disponível em: <https://doi.org/10.1007/s11069-021-05112-x>.
- ALLEN, E. B.; KHOSHGOFTAAR, T. M.; CHEN, Y. Measuring coupling and cohesion of software modules: An information-theory approach. *International Software Metrics Symposium, Proceedings*, n. 561, p. 124–134, 2001.
- ALVES, R. et al. Redes veiculares: Princípios, aplicações e desafios. In: \_\_\_\_\_. [S.l.: s.n.], 2009. p. 199–254.

ALVES, R. d. S. et al. Uma análise experimental da capacidade de redes ad hoc veiculares. *SBrTSimpósio Brasileiro de Telecomunicações*, p. 8, 2008.

AMELIA, A.; SAPTAWATI, G. A. P. Detection of potential traffic jam based on traffic characteristic data analysis. In: *2014 International Conference on Data and Software Engineering (ICODSE)*. [S.l.: s.n.], 2014. p. 1–5.

ANGEL, A. M.; BARTOLO, G. J.; ERNESTINA, M. Predicting recurring concepts on data-streams by means of a meta-model and a fuzzy similarity function. *Expert Systems with Applications*, v. 46, p. 87–105, 2016. ISSN 09574174.

ANJUM, N.; BADRUDDIN, N.; DRIEBERG, M. Simulation of traffic congestion detection using vanets. In: *2014 5th International Conference on Intelligent and Advanced Systems (ICIAS)*. [S.l.: s.n.], 2014. p. 1–4.

ANKERST, M. et al. Optics: Ordering points to identify the clustering structure. *SIGMOD Rec.*, Association for Computing Machinery, New York, NY, USA, v. 28, n. 2, p. 49–60, jun 1999. ISSN 0163-5808. Disponível em: (<https://doi.org/10.1145/304181.304187>).

BABCOCK, M. A. et al. Effects of respiratory muscle unloading on exercise-induced diaphragm fatigue. *J. Appl. Physiol.*, American Physiological Society, v. 93, n. 1, p. 201–206, jul. 2002.

BASAGNI, S. et al. The next paradigm shift: From vehicular networks to vehicular clouds. In: *Mobile Ad Hoc Networking: The Cutting Edge Directions*. [S.l.: s.n.], 2013. p. 645–700.

BASTUG, E.; BENNIS, M.; DEBBAH, M. Living on the edge: The role of proactive caching in 5g wireless networks. *IEEE Communications Magazine*, v. 52, n. 8, p. 82–89, Aug 2014. ISSN 0163-6804.

BATTY, M. et al. Smart cities of the future. *The European Physical Journal Special Topics*, Springer-Verlag, v. 214, n. 1, p. 481–518, nov 2012.

BAUZA, R.; GOZÁLVEZ, J. Traffic congestion detection in large-scale scenarios using vehicle-to-vehicle communications. *Journal of Network and Computer Applications*, Elsevier, v. 36, n. 5, p. 1295–1307, 2013.

BAUZA, R.; GOZALVEZ, J.; SANCHEZ, J. Road traffic congestion detection through cooperative vehicle-to-vehicle communications. In: IEEE. *IEEE Local Computer Network Conference*. [S.l.], 2010. p. 606–612.

BENABDALLAH, F.; HAMZA, A.; BECHRIF, M. Simulation and analysis of vanets performances based on the choice of mobility model. In: IEEE. *Computing Conference, 2017*. [S.l.], 2017. p. 1238–1242.

- BILAL, K. et al. Potentials, trends, and prospects in edge technologies: Fog, cloudlet, mobile edge, and micro data centers. *Computer Networks*, v. 130, p. 94 – 120, 2018. ISSN 1389-1286.
- BOARD, T. R. *Highway Capacity Manual*. [S.l.]: Transportation Research Board, 1997.
- BOARD, T. R. Highway capacity manual. *National Research Council*, 2010.
- BOCK, F. et al. *An Algorithm (the R-th Best Path Algorithm) for Finding and Ranking Paths Through a Network*. Armour Research Foundation Technology Center, 1957. Disponível em: <https://books.google.com.br/books?id=rluJYgEACAAJ>.
- BONOMI, F. et al. Fog computing and its role in the internet of things. In: *Proceedings of the First Edition of the MCC Workshop on Mobile Cloud Computing*. New York, NY, USA: ACM, 2012. (MCC '12), p. 13–16.
- CAETANO, L. de L.; SOARES, A. Redes veiculares: Tendências e estudo de caso. 10 2016.
- CAMPELLO, R. J. G. B.; MOULAVI, D.; SANDER, J. Density-based clustering based on hierarchical density estimates. In: PEI, J. et al. (Ed.). *Advances in Knowledge Discovery and Data Mining*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2013. p. 160–172. ISBN 978-3-642-37456-2.
- CAMPELLO, R. J. G. B. et al. Hierarchical density estimates for data clustering, visualization, and outlier detection. *ACM Trans. Knowl. Discov. Data*, Association for Computing Machinery, New York, NY, USA, v. 10, n. 1, jul 2015. ISSN 1556-4681. Disponível em: <https://doi.org/10.1145/2733381>.
- CASSIANO, K. M. *Análise de séries temporais usando análise espectral singular (ssa) e clusterização de suas componentes baseada em densidade*. Tese (Doutorado), 2015.
- CERQUEIRA, V. et al. On evaluating floating car data quality for knowledge discovery. *IEEE Transactions on Intelligent Transportation Systems*, p. 1–12, 2018. ISSN 1524-9050.
- CHAKRABORTY, R. S. S. Multi-access edge computing for urban informatics. In: . [S.l.]: Association for Computing Machinery, 2022. p. 225–228. ISBN 9781450395601.
- CHATRAPATHI, C. et al. Vanet based integrated framework for smart accident management system. In: IEEE. *2015 International Conference on Soft-Computing and Networks Security (ICSNS)*. [S.l.], 2015. p. 1–7.
- CHEN, K.; LI, Z. Prediction of traffic state based on fuzzy logic in vanet. *Information Technology Journal*, v. 12, n. 18, p. 4642–4646, 2013.
- CHENG, J.; LIU, B.; TANG, X. A traffic-congestion detection method for bad weather based on traffic video. In: LI, K. et al. (Ed.). *Computational Intelligence and Intelligent Systems*. Singapore: Springer Singapore, 2016. p. 506–518. ISBN 978-981-10-0356-1.

Chiti, F.; Fantacci, R.; Rigazzi, G. A mobility driven joint clustering and relay selection for IEEE 802.11p/wave vehicular networks. In: *2014 IEEE International Conference on Communications (ICC)*. [S.l.: s.n.], 2014. p. 348–353.

CIVICWELL. *Curbing Post-Pandemic Traffic*. 2021. <https://www.lgc.org/newsletter/curbing-post-pandemic-traffic/>. Accessed: 2021-09-14.

CYGANEK, B. Change detection in multidimensional data streams with efficient tensor subspace model. In: . [S.l.]: Springer Verlag, 2018. v. 10870 LNAI, p. 694–705. ISBN 9783319926384. ISSN 16113349.

CÁRDENAS-BENÍTEZ, N. et al. Traffic congestion detection system through connected vehicles and big data. *Sensors*, MDPI AG, v. 16, n. 5, p. 599, Apr 2016. ISSN 1424-8220.

D'ANDREA, E.; MARCELLONI, F. Detection of traffic congestion and incidents from gps trace analysis. *Expert Systems with Applications*, v. 73, p. 43 – 56, 2017. ISSN 0957-4174.

DESAI, P. V. A survey on big data applications and challenges. In: *2018 Second International Conference on Inventive Communication and Computational Technologies (ICICCT)*. [S.l.: s.n.], 2018. p. 737–740.

DIMRI, A. et al. Barosense: Using barometer for road traffic congestion detection and path estimation with crowdsourcing. *ACM Trans. Sen. Netw.*, Association for Computing Machinery, New York, NY, USA, v. 16, n. 1, nov. 2019. ISSN 1550-4859.

DING, R. et al. Yading: fast clustering of large-scale time series data. *Proceedings of the VLDB Endowment*, VLDB Endowment, v. 8, n. 5, p. 473–484, 2015.

DREYFUS, S. E. An appraisal of some shortest-path algorithms. *Operations Research, INFORMS*, v. 17, n. 3, p. 395–412, 1969. ISSN 0030364X, 15265463. Disponível em: <http://www.jstor.org/stable/168375>.

EL-SONBATY, Y.; SAID, H. Enhanced density based algorithm for clustering large datasets. In: \_\_\_\_\_. *Computer Recognition Systems 3*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2009. p. 195–203. ISBN 978-3-540-93905-4. Disponível em: [https://doi.org/10.1007/978-3-540-93905-4\\_24](https://doi.org/10.1007/978-3-540-93905-4_24).

ELBAMBY, M. S.; BENNIS, M.; SAAD, W. Proactive edge computing in latency-constrained fog networks. *CoRR*, abs/1704.06749, 2017.

ELBAMBY, M. S. et al. Content-aware user clustering and caching in wireless small cell networks. *CoRR*, abs/1409.3413, 2014.

ESTER, M. et al. A density-based algorithm for discovering clusters in large spatial databases with noise. In: *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining*. [S.l.]: AAAI Press, 1996. (KDD'96), p. 226–231.

- FILIPE, A.; MOURA, N. *User Chat Clustering Using Deep Learning Representations and Unsupervised Methods for Dialog System Applications*. Tese (Doutorado), 2021.
- GALINDO, E.; CORREIA, V.; NETO, L. *A MOBILIDADE URBANA NO BRASIL: PERCEPÇÕES DE SUA POPULAÇÃO*. 2019.
- GAMA, J.; (EDS), M. G. *Learning from Data Streams – Processing techniques in Sensor Networks*. [S.l.]: Springer, 2007.
- GASMI, R.; ALIOUAT, M. Vehicular ad hoc networks versus internet of vehicles - a comparative view. In: *2019 International Conference on Networking and Advanced Systems (ICNAS)*. [S.l.: s.n.], 2019. p. 1–6.
- GERTRUDES, J. C. *Semi-supervised learning approaches with applications in Medicinal Chemistry*. Tese (Doutorado) — Instituto de Ciências Matemáticas e de Computação-USP, São Carlos, Brasil, 5 2019.
- GIACOUMIDIS, E. et al. A Blind Nonlinearity Compensator Using DBSCAN Clustering for Coherent Optical Transmission Systems. *Applied Sciences*, v. 9, n. 20, p. 4398, oct 2019.
- GINO, A. *Highway Traffic Act Investigations*. [S.l.]: Emond Publishing, 2006.
- GOTTHILF, Z.; LEWENSTEIN, M. Improved algorithms for the k simple shortest paths and the replacement paths problems. *Information Processing Letters*, v. 109, n. 7, p. 352–355, 2009. ISSN 0020-0190. Disponível em: <https://www.sciencedirect.com/science/article/pii/S002001900800361X>.
- GUEDES, B. F.; CAMPOS, C. A. A data aggregation scheme for traffic information systems in urban vanets. In: IEEE. *2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC)*. [S.l.], 2016. p. 564–569.
- GUPTA, S. et al. Delay-tolerant and prioritized batch verification system using efficient rsu scheduling in vanet. In: *ICC 2022 - IEEE International Conference on Communications*. [S.l.: s.n.], 2022. p. 5706–5711.
- HARDING, J. et al. *Vehicle-to-Vehicle Communications: Readiness of V2V Technology for Application*. 2014. Disponível em: <https://rosap.ntl.bts.gov/view/dot/27999>.
- HE, Z.; ZHANG, D. Cost-efficient traffic-aware data collection protocol in vanet. *Ad Hoc Networks*, Elsevier, v. 55, p. 28–39, 2017.
- HENRY, X. et al. On-line traffic signal control scheme with real-time delay estimation technology. In: *California partners for advanced transit and highways (PATH). Working Papers: Paper UCB-ITS-PWP-2001*. [S.l.: s.n.], 2001.
- HOFFMAN, W.; PAVLEY, R. A method for the solution of the nth best path problem. *J. ACM*, Association for Computing Machinery, New York, NY, USA, v. 6, n. 4, p. 506–514, oct 1959. ISSN 0004-5411. Disponível em: <https://doi.org/10.1145/320998.321004>.

HOU, X. et al. Vehicular Fog Computing: A Viewpoint of Vehicles as the Infrastructures. *IEEE Transactions on Vehicular Technology*, v. 65, n. 6, p. 3860–3873, 2016.

IDC. *Worldwide Internet of Things Forecast Update 2015–2019*, document #US40983216. Framingham, MA, USA: [s.n.], 2016.

IDC. *Worldwide Internet of Things Forecast Update 2018–2025*, document #US45213219. Framingham, MA, USA: [s.n.], 2019.

IEEE. Ieee standard for telecommunications and information exchange between systems - lan/man specific requirements - part 11: Wireless medium access control (mac) and physical layer (phy) specifications: High speed physical layer in the 5 ghz band. *IEEE Std 802.11a-1999*, p. 1–102, Dec 1999. ISSN null.

IEEE. Ieee standard for information technology– local and metropolitan area networks– specific requirements– part 11: Wireless lan medium access control (mac) and physical layer (phy) specifications amendment 6: Wireless access in vehicular environments. *IEEE Std 802.11p-2010 (Amendment to IEEE Std 802.11-2007 as amended by IEEE Std 802.11k-2008, IEEE Std 802.11r-2008, IEEE Std 802.11y-2008, IEEE Std 802.11n-2009, and IEEE Std 802.11w-2009)*, p. 1–51, July 2010. ISSN null.

IEEE. Ieee standard for wireless access in vehicular environments (wave)–identifiers. *IEEE Std 1609.12-2019 (Revision of IEEE Std 1609.12-2016)*, p. 1–17, Oct 2019. ISSN null.

INFOMONEY. *InfoMoney*. 2022.

IORGA, M. et al. *Fog computing conceptual model*. 2018. Disponível em: <http://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.500-325.pdf>.

IRREVALDY; SAPTAWATI, G. A. P. Spatio-temporal mining to identify potential traffic congestion based on transportation mode. In: *2017 International Conference on Data and Software Engineering (ICoDSE)*. [S.l.: s.n.], 2017. p. 1–6.

ITDP. *The Next Pandemic Surge: Traffic*. 2021. <https://www.itdp.org/2021/03/22/the-next-pandemic-surge-traffic/>. Accessed: 2021-09-14.

JĘDRZEJOWICZ, J.; JĘDRZEJOWICZ, P. Distance-based ensemble online classifier with kernel clustering. *Smart Innovation, Systems and Technologies*, v. 39, p. 279–289, 2015. ISSN 21903026.

KAUR, K.; KAD, S. Enhanced clustering based aodv-r protocol using ant colony optimization in vanets. In: IEEE. *2016 IEEE 1st International Conference on Power Electronics, Intelligent Control and Energy Systems (ICPEICES)*. [S.l.], 2016. p. 1–5.

Kerner, B. S. et al. Traffic state detection with floating car data in road networks. In: *Proceedings. 2005 IEEE Intelligent Transportation Systems, 2005*. [S.l.: s.n.], 2005. p. 44–49. ISSN 2153-0017.

KRAWCZYK, B.; WOŹNIAK, M. One-class classifiers with incremental learning and forgetting for data streams with concept drift. *Soft Computing*, v. 19, n. 12, p. 3387–3400, 2015. ISSN 14337479.

KUMAR, T.; KUSHWAHA, D. S. An efficient approach for detection and speed estimation of moving vehicles. *Procedia Comput. Sci*, v. 89, p. 726–731, 2016.

KUMAR, T.; KUSHWAHA., D. S. An approach for traffic congestion detection and traffic control system. In: . [S.l.]: Information and communication technology for competitive strategies, Springer, 2019. p. 99–108.

KYKUTA, D. H. *Comparação de algoritmos para o Problema dos K Menores Caminhos*. 2018.

LAI, C. et al. Effective density queries for moving objects in road networks. In: DONG, G. et al. (Ed.). *Advances in Data and Web Management*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2007. p. 200–211. ISBN 978-3-540-72524-4.

LAM, C.; GAO, H.; NG, B. A real-time traffic congestion detection system using on-line images. In: *2017 IEEE 17th International Conference on Communication Technology (ICCT)*. [S.l.: s.n.], 2017. p. 1548–1552.

LI, Y.; WANG, H.; BUCKLES, B. Traffic congestion assessment based on street level data for on-edge deployment. In: *Proceedings of the 4th ACM/IEEE Symposium on Edge Computing*. New York, NY, USA: Association for Computing Machinery, 2019. (SEC '19), p. 289–291. ISBN 9781450367332.

LI, Y. J. An overview of the dsrc/wave technology. In: ZHANG, X.; QIAO, D. (Ed.). *Quality, Reliability, Security and Robustness in Heterogeneous Networks*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012. p. 544–558.

LIMA, A. M. S.

*Monitoramento de transições em agrupamento de fluxos de dados*, 2022.

LV, M. et al. Detecting traffic congestions using cell phone accelerometers. In: *Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing: Adjunct Publication*. New York, NY, USA: Association for Computing Machinery, 2014. (UbiComp '14 Adjunct), p. 107–110. ISBN 9781450330473.

MANJORO, W. S.; DHAKAR, M.; CHAURASIA, B. K. Traffic congestion detection using data mining in vanet. In: *2016 IEEE Students' Conference on Electrical, Electronics and Computer Science (SCEECS)*. [S.l.: s.n.], 2016. p. 1–6.

MCINNES, L.; J, H.; S., A. *How HDBSCAN Works*. 2016. Último acesso em 10 junho 2022. Disponível em: <[https://hdbscan.readthedocs.io/en/latest/how\\_hdbscan\\_works.html](https://hdbscan.readthedocs.io/en/latest/how_hdbscan_works.html)>.

MENG, C. et al. The 11th international workshop on urban computing. In: . [S.l.]: Association for Computing Machinery (ACM), 2022. p. 4886–4887. ISBN 9781450393850.

MERAIHI, R. et al. Vehicle-to-vehicle communications: Applications and perspectives. In: \_\_\_\_\_. *Wireless Ad Hoc and Sensor Networks*. John Wiley Sons, Ltd, 2010. cap. 12, p. 285–308. ISBN 9780470610893. Disponível em: <https://onlinelibrary.wiley.com/doi/abs/10.1002/9780470610893.ch12>.

MOHANTY, A. et al. Identification and evaluation of the effective criteria for detection of congestion in a smart city. *IET Communications*, v. 16, n. 5, p. 560–570, 2022. Disponível em: <https://ietresearch.onlinelibrary.wiley.com/doi/abs/10.1049/cmu2.12344>.

MUKHERJEE, M.; SHU, L.; WANG, D. Survey of Fog Computing: Fundamental, Network Applications, and Research Challenges. *IEEE Communications Surveys & Tutorials*, IEEE, v. 20, n. 3, p. 1–1, 2018. ISSN 1553-877X.

NADEEM, T. et al. Trafficview: traffic data dissemination using car-to-car communication. *ACM SIGMOBILE Mobile Computing and Communications Review*, ACM, v. 8, n. 3, p. 6–19, 2004.

NIARI, A. K.; BERANGI, R.; FATHY, M. Ecn: an extended ccn architecture to improve data access in vehicular content-centric network. *The Journal of Supercomputing*, v. 74, n. 1, p. 205–221, Jan 2018.

NING, Z.; HUANG, J.; WANG, X. Vehicular Fog Computing: Enabling Real-Time Traffic Management for Smart Cities. *IEEE Wireless Communications*, IEEE, v. 26, n. 1, p. 87–93, feb 2019.

NOORI, H.; OLYAEI, B. B. A novel study on beaconing for vanet-based vehicle to vehicle communication: Probability of beacon delivery in realistic large-scale urban area using 802.11p. In: *2013 International Conference on Smart Communications in Network Technologies (SaCoNeT)*. [S.l.: s.n.], 2013. v. 01, p. 1–6.

NUNES, S. A. et al. Finding spatio-temporal patterns in multidimensional data streams. *Journal of Information and Data Management*, p. 227–340, 2013.

OLIVEIRA, S. S.; KNISS, J. Redução de dados em redes de sensores sem fio baseada em análise de dispersão. In: SBC. *Anais do XXXVII Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos*. [S.l.], 2019. p. 1–14.

OMNET++. *Omnet++ - Network Simulation Framework*. Available: <https://www.omnetpp.org/>. Access: 15/04/2020. [S.l.: s.n.], 2019.

PALUBINSKAS, G.; KURZ, F.; REINARTZ, P. Detection of traffic congestion in optical remote sensing imagery. In: *IGARSS 2008 - 2008 IEEE International Geoscience and Remote Sensing Symposium*. [S.l.: s.n.], 2008. v. 2, p. II–426–II–429.

PATROUMPAS, K.; SELLIS, T. Window specification over data streams. In: SPRINGER. *International Conference on Extending Database Technology*. [S.l.], 2006. p. 445–464.

PEIXOTO, M.; GENEZ, T.; BITTENCOURT, L. F. Hierarchical scheduling mechanisms in multi-level fog computing. *IEEE Transactions on Services Computing*, p. 1–1, 2022.

PEIXOTO, M. et al. A traffic data clustering framework based on fog computing for vanets. *Vehicular Communications*, v. 31, p. 100370, 2021. ISSN 2214-2096. Disponível em: <https://www.sciencedirect.com/science/article/pii/S2214209621000395>).

PEIXOTO, M. et al. Fogjam: A fog service for detecting traffic congestion in a continuous data stream vanet. *Ad Hoc Networks*, v. 140, p. 103046, 2023. ISSN 1570-8705. Disponível em: <https://www.sciencedirect.com/science/article/pii/S1570870522002189>).

PEIXOTO, M. L. M. et al. Exploiting fog computing with an adapted dbscan for traffic congestion detection system. In: *2020 IEEE 92nd Vehicular Technology Conference (VTC2020-Fall)*. [S.l.: s.n.], 2020. p. 1–5.

PELLEG, D.; MOORE, A. W. X-means: Extending k-means with efficient estimation of the number of clusters. In: *Proceedings of the Seventeenth International Conference on Machine Learning*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2000. (ICML '00), p. 727–734. ISBN 1558607072.

RAMACHANDRA, S. H. et al. A novel dynamic traffic management system using on board diagnostics and zigbee protocol. In: *2016 International Conference on Communication and Electronics Systems (ICCES)*. [S.l.: s.n.], 2016. p. 1–6.

RANI, P.; SHAW, D. K. A hybrid approach for traffic delay estimation. In: KRISHNA, A.; SRIKANTIAH, K.; NAVEENA, C. (Ed.). *Integrated Intelligent Computing, Communication and Security*. Singapore: Springer Singapore, 2019. p. 243–250. ISBN 978-981-10-8797-4.

RAO, A. et al. Dynamic traffic system based on real time detection of traffic congestion. In: *2018 Fourth International Conference on Computing Communication Control and Automation (ICCUBEA)*. [S.l.: s.n.], 2018. p. 1–5.

RATH, M.; PATI, B.; PATTANAYAK, B. K. Mobile agent-based improved traffic control system in vanet. In: KRISHNA, A.; SRIKANTIAH, K.; NAVEENA, C. (Ed.). *Integrated Intelligent Computing, Communication and Security*. Singapore: Springer Singapore, 2019. p. 261–269. ISBN 978-981-10-8797-4.

RENDA, M. E. et al. Ieee 802.11 p vanets: Experimental evaluation of packet inter-reception time. *Computer Communications*, Elsevier, v. 75, p. 26–38, 2016.

RODRIGUEZ, E.; FERREIRA, N.; POCO, J. JamVis: exploration and visualization of traffic jams. *The European Physical Journal Special Topics*, Springer Science

and Business Media LLC, jan. 2022. Disponível em: [⟨https://doi.org/10.1140/epjs/s11734-021-00424-2⟩](https://doi.org/10.1140/epjs/s11734-021-00424-2).

ROUSSEEUW, P. Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. *J. Comput. Appl. Math.*, Elsevier Science Publishers B. V., NLD, v. 20, n. 1, p. 53–65, nov 1987. ISSN 0377-0427. Disponível em: [⟨https://doi.org/10.1016/0377-0427\(87\)90125-7⟩](https://doi.org/10.1016/0377-0427(87)90125-7).

SADRI, A. A. et al. Data reduction in fog computing and internet of things: A systematic literature survey. *Internet of Things*, v. 20, p. 100629, 11 2022. ISSN 25426605. Disponível em: [⟨https://linkinghub.elsevier.com/retrieve/pii/S254266052200110X⟩](https://linkinghub.elsevier.com/retrieve/pii/S254266052200110X).

SARKAR, S.; CHATTERJEE, S.; MISRA, S. Assessment of the Suitability of Fog Computing in the Context of Internet of Things. *IEEE Transactions on Cloud Computing*, v. 6, n. 1, p. 46–59, jan 2018.

SATYANARAYANA, G. S. R.; MAJHI, S.; DAS, S. K. A vehicle detection technique using binary images for heterogeneous and lane-less traffic. *IEEE Transactions on Instrumentation and Measurement*, v. 70, p. 1–14, 2021.

SCHMIDT, R. K. et al. Exploration of adaptive beaconing for efficient intervehicle safety communication. *IEEE Network*, v. 24, n. 1, p. 14–19, Jan 2010.

SCHRANK, D. et al. Urban mobility report 2021. *Published by The Texas A&M Transportation Institute with cooperation from INRIX Research*, 2021.

SCHWARZ, G. Estimating the dimension of a model. *The Annals of Statistics*, Institute of Mathematical Statistics, v. 6, n. 2, p. 461–464, 1978. ISSN 00905364.

SHAH, Y. A. et al. An evolutionary algorithm-based vehicular clustering technique for vanets. *IEEE Access*, v. 10, p. 14368–14385, 2022.

SHAW, B. A.; BELTRÁN, H. F.; SOWERBY, K. W. *Valuing spectrum at mm wavelengths for cellular networks*. [S.l.], 2017. Disponível em: [⟨https://ideas.repec.org/p/zbw/itsp17/168539.html⟩](https://ideas.repec.org/p/zbw/itsp17/168539.html).

SHEN, J. et al. Secure real-time traffic data aggregation with batch verification for vehicular cloud in vanets. *IEEE Transactions on Vehicular Technology*, IEEE, v. 69, n. 1, p. 807–817, 2019.

SILVA, A. E. D. et al. Halite ds : Fast and scalable subspace clustering for multidimensional data streams \*. In: . [s.n.], 2016. p. 351–359. Disponível em: [⟨https://epubs.siam.org/terms-privacy⟩](https://epubs.siam.org/terms-privacy).

SILVA, J. A. et al. Data stream clustering: A survey. *ACM Computing Surveys*, v. 46, 10 2013. ISSN 03600300.

SILVA, T. L. C. da et al. Efficient and distributed dbscan algorithm using mapreduce to detect density areas on traffic data. In: *ICEIS (1)*. [S.l.: s.n.], 2014. p. 52–59.

- SOUSA, E. P. M. de et al. Measuring evolving data streams' behavior through their intrinsic dimension. In: . [S.l.]: Ohmsha, Ltd. and Springer, 2006. v. 25, p. 33–59.
- SOUZA, A. M. de et al. Itssafe: An intelligent transportation system for improving safety and traffic efficiency. In: *2018 IEEE 87th Vehicular Technology Conference (VTC Spring)*. [S.l.: s.n.], 2018. p. 1–7. ISSN 2577-2465.
- SOUZA, A. M. de; VILLAS, L. A. A fully-distributed traffic management system to improve the overall traffic efficiency. In: *Proceedings of the 19th ACM International Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems*. New York, NY, USA: ACM, 2016. (MSWiM '16), p. 19–26. ISBN 978-1-4503-4502-6.
- SOUZA, A. M. de et al. Icarus: Improvement of traffic condition through an alerting and re-routing system. *Computer Networks*, Elsevier, v. 110, p. 118–132, 2016.
- SPINOSA, E. J. *Deteção de novidade com aplicação a fluxos contínuos de dados*. Tese (Doutorado), 2015.
- SUMO. *SUMO - Simulation of Urban Mobility*. Available: < [http : //sumo.sourceforge.net/](http://sumo.sourceforge.net/) >. Access: 15/04/2020. [S.l.: s.n.], 2019.
- TANIZAKI, M.; WOLFSON, O. Randomization in traffic information sharing systems. In: *Proceedings of the 15th Annual ACM International Symposium on Advances in Geographic Information Systems*. New York, NY, USA: Association for Computing Machinery, 2007. (GIS '07). ISBN 9781595939142.
- TATBUL, N. et al. Load shedding in data stream manager. In: *Proceedings - 29th International Conference on Very Large Data Bases, VLDB 2003*. [S.l.]: Elsevier, 2003. p. 309–320.
- THILAGAMANI, S.; JAYANTHILADEVI, A.; ARUNKUMAR, N. Data mining algorithms, fog computing. In: *Handbook of Research on Cloud and Fog Computing Infrastructures for Data Science*. IGI Global, 2018. p. 231–264. Disponível em: <<https://doi.org/10.4018/978-1-5225-5972-6.ch012>>.
- TIMES, L. *L.A. traffic behavior is changing. Is post-pandemic gridlock inevitable?* 2021. <<https://www.latimes.com/california/story/2021-06-18/post-covid-l-a-traffic-analysis-has-rush-hour-changed>>. Accessed: 2021-09-14.
- TOLPIN, D. Progressive temporal window widening. *arXiv preprint arXiv:1604.00997*, 2016.
- TSUYOSHI, I. et al. City-wide traffic flow estimation from a limited number of low-quality cameras. *IEEE Transactions on Intelligent Transportation Systems*, v. 18, n. 4, p. 950–959, April 2017. ISSN 1558-0016.
- UPPOOR, S. et al. Generation and analysis of a large-scale urban vehicular mobility dataset. *IEEE Transactions on Mobile Computing*, v. 13, n. 5, p. 1061–1075, 2014.

VEGNI, A. M.; LITTLE, T. D. Hybrid vehicular communications based on v2v-v2i protocol switching. *International Journal of Vehicle Information and Communication Systems*, Inderscience Publishers, v. 2, n. 3-4, p. 213–231, 2011.

VEINS. *Vehicles In Network Simulation*. Available: < <http://veins.car2x.org> >. Access: 15/04/2020. [S.l.: s.n.], 2019.

VITTER, J. S. Random sampling with a reservoir. *ACM Transactions on Mathematical Software (TOMS)*, v. 11, n. 1, p. 37–57, mar 1985.

WAHID, A.; RAO, A. C. S.; GOEL, D. Server communication reduction for gps-based floating car data traffic congestion detection method. In: KRISHNA, A.; SRIKANTIAH, K.; NAVEENA, C. (Ed.). *Integrated Intelligent Computing, Communication and Security*. Singapore: Springer Singapore, 2019. p. 415–425. ISBN 978-981-10-8797-4.

WANG, R.; WORK, D. B.; SOWERS, R. Multiple model particle filter for traffic estimation and incident detection. *IEEE Transactions on Intelligent Transportation Systems*, v. 17, n. 12, p. 3461–3470, 2016.

WANG, S. et al. Efficient traffic estimation with multi-sourced data by parallel coupled hidden markov model. *IEEE Transactions on Intelligent Transportation Systems*, v. 20, n. 8, p. 3010–3023, 2019.

WANG, Y. et al. Twitter-based traffic delay detection based on topic propagation analysis using railway network topology. *Personal and Ubiquitous Computing*, Springer, v. 23, n. 2, p. 233–247, 2019.

WU, X. et al. Vehicular Communications Using DSRC: Challenges, Enhancements, and Evolution. *IEEE Journal on Selected Areas in Communications*, v. 31, n. 9, p. 399–408, sep 2013.

XU, L.; YUE, Y.; LI, Q. Identifying urban traffic congestion pattern from historical floating car data. *Procedia - Social and Behavioral Sciences*, v. 96, p. 2084 – 2095, 2013. ISSN 1877-0428. Intelligent and Integrated Sustainable Multimodal Transportation Systems Proceedings from the 13th COTA International Conference of Transportation Professionals (CICTP2013).

XU, W. et al. Internet of vehicles in big data era. *IEEE/CAA Journal of Automatica Sinica*, v. 5, n. 1, p. 19–35, 2018.

YEN, J. Y. Finding the k shortest loopless paths in a network. *Management Science, INFORMS*, v. 17, n. 11, p. 712–716, 1971. ISSN 00251909, 15265501. Disponível em: <<http://www.jstor.org/stable/2629312>>.

YOUNAS, I.; RAUF, A.; SYED, W. Towards traffic delay estimation using fuzzy logic. In: *International business information management conference Bescia*. [S.l.: s.n.], 2006.

YOUSEFI, S.; MOUSAVI, M. S.; FATHY, M. Vehicular ad hoc networks (vanets): challenges and perspectives. In: IEEE. *2006 6th International Conference on ITS Telecommunications*. [S.l.], 2006. p. 761–766.

ZHANG, S. et al. Representation of traffic congestion data for urban road traffic networks based on pooling operations. *Algorithms*, v. 13, n. 4, 2020. ISSN 1999-4893. Disponível em: <https://www.mdpi.com/1999-4893/13/4/84>.

ZHANG, Y. et al. A method for traffic congestion clustering judgment based on grey relational analysis. *ISPRS International Journal of Geo-Information*, MDPI AG, v. 5, n. 5, p. 71, May 2016. ISSN 2220-9964.

ZHENG, J.; GAO, D. W.; LIN, L. Smart meters in smart grid: An overview. In: *2013 IEEE Green Technologies Conference (GreenTech)*. [S.l.: s.n.], 2013. p. 57–64. ISSN 2166-546X.

ZHENG, Y. et al. Urban computing: Concepts, methodologies, and applications. *ACM Trans. Intell. Syst. Technol.*, ACM, New York, NY, USA, v. 5, n. 3, p. 38:1–38:55, set. 2014. ISSN 2157-6904.