

**A REAL-TIME DYNAMIC FOG NODE SELECTION
TECHNIQUE TO ENHANCE QOS AND REDUCE
ENERGY CONSUMPTION IN VEHICULAR FOG
NETWORK**

A thesis submitted to
UPES

For the award of
Doctor of Philosophy

In
Computer Science and Engineering

BY
Deep Chandra Binwal

May 2024

SUPERVISOR(s)

Dr. Shamik Tiwari

Dr. Monit Kapoor



School of Computer Science

UPES

Dehradun-248007: Uttarakhand

**A REAL-TIME DYNAMIC FOG NODE SELECTION
TECHNIQUE TO ENHANCE QOS AND REDUCE
ENERGY CONSUMPTION IN VEHICULAR FOG
NETWORK**

A thesis submitted to

UPES

For the award of

Doctor of Philosophy

in

Computer Science and Engineering

BY

Deep Chandra Binwal

(SAP ID: 500078646)

May 2024

SUPERVISOR(s)

Dr. Shamik Tiwari

(Prof. SCSE, IILM University Gurugram)

(Ex Prof. School of Computer Science, UPES)

Dr. Monit Kapoor

(Pro VC & Director ASET, Amity University, Bengaluru)



School of Computer Science

UPES

Dehradun-248007: Uttarakhand

Declaration

I declare that the thesis entitled "*A Real-time Dynamic Fog Node Selection Technique to Enhance QoS and Reduce Energy Consumption in Vehicular Fog Network*" has been prepared by me under the guidance of Dr. Shamik Tiwari, Professor, SCSE, IILM University, Gurugram (Ex Prof. SCS, UPES, Dehradun) and Dr. Monit Kapoor, Pro Vice Chancellor & Director ASET, Amity University, Bengaluru. No part of this thesis has formed the basis for the award of any degree or fellowship previously.



Deep Chandra Binwal
School of Computer Science
UPES, Dehradun

Certificate

I certify that Deep Chandra Binwal has prepared his thesis entitled “A Real-time Dynamic Fog Node Selection Technique to Enhance QoS and Reduce Energy Consumption in Vehicular Fog Network”, for award of PhD degree of the UPES, under our guidance. He has carried out the work at the School of Computer Science, UPES, Dehradun, India.

Supervisor



Dr Shamik Tiwari

Professor

School of Computer Science & Engineering

IILM University, Gurugram

Ex-Prof.- UPES, Dehradun



AMITY
UNIVERSITY
BENGALURU

CERTIFICATE

We certify that Deep Chandra Binwal has prepared his thesis entitled "A Real-time Dynamic Fog Node Selection Technique to Enhance QoS and Reduce Energy Consumption in Vehicular Fog Network", for award of PhD degree of the UPES, under our guidance. He has carried out the work at the School of Computer Science, UPES, Dehradun, India.

Co - Supervisor

Monit Kapoor

Dr. Monit Kapoor

Pro Vice Chancellor and Director

Amity School of Engineering and Technology

Amity University, Bengaluru

Date : May 2024



Abstract

Connected vehicles are helping realize the dream of intelligent transportation system (ITS) a come true. With the steady rise in the on-board processing ability of vehicles, demand for new services and applications is increasing. The majority of such applications are latency-sensitive and have safety-related consequences. The latency is of critical importance for ensuring safety and quality of service (QoS) in ITS environment. However, achieving low latency, while meeting the high workload demand from vehicular nodes in Vehicular fog computing (VFC) is challenging due to dynamic changes in resource demand and availability. VFC is an on-demand collaboration of on-board vehicular computational resources to enhance the capability of fog servers. It is an important idea to solve various issues related to fog computing such as, high infrastructure cost, sub-optimal utilization of computational resources, and rise of many new applications, etc. VFC is the enabler of real-time, safety-critical, and latency-sensitive applications in ITS. VFC makes computing power available much closer to the vehicles and end-user thereby, reducing latency. For safety-critical applications meeting maximum permissible latency is most important due to the temporal relevance of results of queries. Although, VFC is a very promising field of study, it has some important unresolved research challenges as well. The VFC research area has seen various architectural, application and services, and implementation issues, researched by academia. The VFC is a resource constraint environment thus, optimal utilization of resources along with meeting latency limits is the most important challenge in VFC.

In this research work, the architecture of VFC has been studied bringing out several unique features of VFC and providing a comparison with conventional fog computing. The effect of VFC on ITS considering architecture, applications and services, and challenges has been analyzed. Existing applications and research issues in them are presented, bringing out a new application use case of vehicular fog computing, "bridging cellular network coverage holes using vehicular fog computing". This research work also presents new and promising applications in VFC along with depicting open research challenges. A research investigation is conducted on characteristics of optimal Vehicular Fog Node (VFN), and a multi-variable constrained non-linear multi-objective optimization (MOO) model-based technique is proposed, for minimizing latency in vehicular fog network. Then proposed MOO technique is evaluated for optimum latency under varying parameters like radio bandwidth, the number of vehicles, request arrival rate, and workload sizes. Statistical validation of simulation data is conducted using a one-way ANOVA test, with 95 percentile confidence. The effect of vehicle fleet size, workload size, request arrival rate, and probability of processing at VFN is analyzed for the latency and QoS satisfaction for real-time, latency-sensitive requests in VFC. The proposed technique is compared with generic fog-cloud, *ECOS*, and *ETEFN* techniques for latency and QoS satisfaction. Simulation results depict on average 350 times higher latency for workload sizes from 300 to 1000, for the generic fog-cloud system as compared to the proposed VFC model. The QoS satisfaction ratio is maintained between 96 and 93 percent for the proposed technique as compared to 93 to 84 percent for *ECOS*, and 94 to 78 percent for *ETEFN* for the request arrival rate from 0.5 to 5.0. Whereas, the QoS satisfaction ratio declines very sharply from 88 to 10 percent for generic fog-cloud system. The proposed MOO technique for VFC in ITS offers quantitative QoS enhancement as compared to the generic fog-cloud, and the state-of-the-art techniques, which makes it suitable for adaptation for VFC in real-time applications.

Further, a multi-objective optimization (MOO) model is presented to optimize the resource utilization and latency in VFC by optimal selection of dynamic fog nodes for processing latency-sensitive requests. A heuristic-based algorithm, lightweight energy-efficient algorithm for quality-of-service (LEAQoS) is pro-

posed and designed to solve the MOO model. A novel concept of adaptive capacity tuning for the dynamic workload (ACT-DW) has been presented for a real-time, and dynamic resource adaption to workload for optimizing the energy consumption, latency, and minimizing cost to fog service providers. In this research work, the processing for proposed algorithms (for scheduling, management, and allocation of resources) is performed at static fog nodes (i.e., FCN), and latency-sensitive request processing is performed at VFN and (or) FCN. Thus, all the VFN resources are dedicated to request processing thus enhanced quality of service (QoS) is achieved. A novel implementation of well known publisher-subscriber communication architecture is presented in this research work for processing of optimal fog node selection algorithms, and allocation of optimal resources to requests. To prove the practical applicability of the proposed technique, the iFogSim2 simulation tool is used with realistic vehicular data from Chandigarh, India, using open street map (OSM) and SUMO. The comparison of the simulation results with previous competing research works proves that latency for the proposed work remains within 10 percent limits, in contrast to a sharp increase in previous works for an increase in the number of VNs from 20 to 100. The energy consumption is on average 21 percent lower for the proposed scheme. The network bandwidth consumption is 27 to 41 percent lower in the current research work when compared to previous works. The percentage of requests satisfying the maximum latency constraint is between 94 and 97 in the current research work when compared to a sharp decrease with an increasing number of VNs beyond 40, in previous competing works. This proves the quantitative enhancement in QoS and energy efficiency in the current proposed research work when compared to the previous competing research works.

Keywords: fog computing, intelligent transportation system (ITS), latency-sensitive, multi-objective optimization (MOO), on-board unit (OBU), vehicular fog computing (VFC), vehicular fog node, quality of service (QoS).

Acknowledgement

I remain indebted to my research supervisor(s) Dr. Shamik Tiwari for his guidance, positive assurance, and valuable suggestions towards the completion of my Ph.D. thesis. I would like to express my deepest gratitude to Dr. Monit Kapoor, my co-supervisor, with whom I started my research journey. His mentorship and guidance in writing my first journal paper will always remain with me as my stepping stone in my academic research journey. He is very thorough in critically analysing the research work, and very generous in assisting to improve the work. I owe a lot to Dr. Rajeev Tiwari, who led my research work to a different level and constantly persuaded me to elevate the level of my research work. He encouraged me to publish my research papers in best of the journals in the field. Because of his consistent encouragement and guidance, I could complete and publish my research work timely. I am grateful to my friends Dr. Narendra Sijwali and Dr. Gopal Joshi from whom I derived my motivation to join the Ph.D. course. I am grateful to Dr. Bhupinder Singh for his motivating words in times of distress during my research journey. I thank the entire team of the School of Computer Science and, the UPES research department for helping me with whatever help I required and being so cordial and responsive during my interaction with them. I am grateful to all my colleagues and teammates at my office in the Indian Air Force for being so supportive of me throughout my Ph.D. research period.

The greatest support during my entire Ph.D. journey was from my beloved wife Kiran, who firstly permitted and then supported me to undertake this am-

bitious mission. Her unhindered love, emotional support, and encouragement enabled me to continue my research, particularly during difficult phases of research. Without her patience, love, encouraging words and support, this would not have been possible. I draw energy and enthusiasm from my twin boys Lucky and Vicky (16 years). They always keep me active, play with me, and motivate me to work hard towards my research goals. Both my sons have always supported my studies and spared me enough time for my research. I am grateful to all my friends and extended family members who provided support and motivation to me in countless ways. Lastly, I would like to acknowledge all the other people who have been of direct or indirect help and assisted me throughout my research. I would like to dedicate this thesis to my mother, Smt Keshvi Binwal, and my late father, Shri Rewadhar Binwal, who have always encouraged me to follow my dream and achieve new heights.

(Deep Chandra Binwal)

Table of Contents

Declaration	ii
Certificate	iii
Abstract	v
Acknowledgement	viii
Table of Contents	x
List of Tables	xiii
List of Figures	xv
List of Symbols and Abbreviation	xvii
s1 Introduction	1
1.1 Cloud Computing	6
1.2 Fog Computing	6
1.3 Vehicular Fog Computing	8
1.3.1 The Benefits of Vehicular Fog Computing	8
1.3.2 Dynamic Fog Nodes in Vehicular Fog Computing	9
1.4 Motivation to Undertake This Research	10
1.5 Problem Statement	11
1.6 Research Objectives	13
2 Literature Review	14
2.1 Introduction	14
2.2 Intelligent Transportation System	14

2.3	Evolution of Vehicular Network Research	15
2.4	Fog Computing Characteristics	15
2.5	Architecture and Applications of Vehicular Fog Computing . . .	17
2.6	Summary of Literature Review	23
2.7	Analysis of the Comprehensive Literature Review	26
2.7.1	Characteristics of Vehicular Fog Computing	28
2.7.2	Comparison of Vehicular Fog Computing and Conventional Fog Computing	31
2.7.3	Application and Services in VFC	36
2.7.4	Applications of VFC	37
2.7.5	New Applications of VFC Identified	40
2.7.6	A Simple VFC Application Use-case Illustrated	42
2.7.7	Classification of Vehicular Network Applications	42
2.8	Research Challenges	43
2.8.1	Summary of Previously Presented Research Challenges .	45
2.8.2	New Research Challenges Identified	47
2.9	Summary	48
3	Multi Objective Modelling of Vehicular Fog Computing	49
3.1	Introduction	49
3.2	System Model Architecture	50
3.3	Optimal Vehicular Fog Node Characteristics	53
3.4	Communication and Propagation Model	54
3.5	Mobility Model	57
3.6	Networking Model	58
3.7	Latency Model	59
3.7.1	Latency Element Definitions	60
3.8	Energy Model	64
3.9	Multi Objective Optimization (MOO) Model	66
3.9.1	Assumptions and Pre-requisites	67
3.9.2	QoS Satisfaction Ratio	67
3.9.3	Multi Objective Optimization Model Definition	67
3.10	Simulation Environment	69
3.11	Simulation Results and Analysis	72
3.12	Summary	77

4	A Lightweight Energy-efficient Technique for QoS Enhancement in Urban VFC for Intelligent Transportation System	79
4.1	Introduction	79
4.2	Motivation to Undertake Research Work	82
4.3	System Model Architecture	83
4.3.1	MOO Problem Formulation	84
4.4	Proposed Solution Technique for Optimal VFN Selection	85
4.4.1	Heuristics-based Algorithm for Solving the Mathematical Model	86
4.5	Important Contributions	92
4.6	Summary	93
5	Performance Evaluation	94
5.1	Introduction	94
5.2	Simulation Environment	95
5.3	Result and discussion	101
5.3.1	Effect on Latency	101
5.3.2	Effect on Energy consumption	102
5.3.3	Effect on Network Bandwidth Consumption	103
5.3.4	Effect on Maximum Latency Satisfaction	104
5.3.5	Statistical Validation of the Simulation Results	105
5.4	Summary	109
6	Conclusion and Future Research Suggestions	110
6.1	Introduction	110
6.2	Summary of Research Findings	111
6.3	Contributions of the Research Work	112
6.4	Limitations and Future Research	113
6.4.1	Limitations of the Sample Data used during Simulation .	113
6.4.2	Evolution of Research	113
	References	115
	List of Publications	133

List of Tables

Table. 2.1	Summary of the systematic literature review	23
Table. 2.2	Differences between conventional fog and VFC	33
Table. 2.3	Similarities between conventional fog and VFC	35
Table. 2.4	Potential applications of vehicular fog computing	37
Table. 2.5	Summary of research challenges previously discussed by authors	45
Table. 3.1	Different symbols used in equation (3.13) and their de- scription	63
Table. 3.2	States of node and corresponding power consumption . .	65
Table. 3.3	Different input arguments used with 'fmincon' solver . .	71
Table. 3.4	Variables and the values of constraints	71
Table. 3.5	Summary of one-way ANOVA test on simulation data . .	72
Table. 3.6	Parameter values used for calculating the QoS satisfaction ratio	74
Table. 4.1	List of latency-sensitive applications in the connected ve- hicle domain	80
Table. 4.2	Symbols used in Algorithms 1-3 and their description . .	86
Table. 5.1	Values of various simulation parameters	98
Table. 5.2	Values of various parameters of request vector	99
Table. 5.3	Different problem configurations	99
Table. 5.4	Standard deviation of the obtained latency values	100
Table. 5.5	Standard deviation of the obtained energy consumption values	100
Table. 5.6	Standard deviation of the obtained network usage values .	101

Table. 5.7	Levene's test summary for latency results	106
Table. 5.8	Levene's test statistics for latency results ($\sigma = 0.05$) . . .	107
Table. 5.9	Levene's test summary for energy consumption results . .	107
Table. 5.10	Levene's test statistics for energy consumption results ($\sigma = 0.05$)	107
Table. 5.11	Levene's test summary for network usage results	108
Table. 5.12	Levene's test statistics for network usage results ($\sigma = 0.05$)	108

List of Figures

Figure. 1.1	Evolution of Research in VFC and ITS	4
Figure. 1.2	Taxonomy of vehicular network application	6
Figure. 1.3	Taxonomy of computational techniques in IoV	7
Figure. 1.4	Architecture of Vehicular Fog Computing	13
Figure. 2.1	Evolution of computing technology	27
Figure. 2.2	Hierarchical architecture of VFC	27
Figure. 2.3	Comparison of the characteristics of vehicular fog, fog, and cloud computing characteristics	32
Figure. 2.4	Comparison of the characteristics of Vehicular Fog and Fog	35
Figure. 2.5	Categories of VFC applications	37
Figure. 2.6	VFC network bridging the coverage gap of cellular network	43
Figure. 2.7	Classification of the vehicular networking applications . .	43
Figure. 3.1	The architecture of VFC network: task processing and despatching to higher layers	52
Figure. 3.2	Flow chart of request processing	57
Figure. 3.3	Plot of the latency variation with workload sizes	73
Figure. 3.4	Variation of QoS satisfaction ratio with request arrival rate	75
Figure. 3.5	Variation in latency with the probability of processing a request at VFN for different fleet sizes (number of VFN = 4)	76
Figure. 3.6	Variation in latency with the probability of processing a request at VFN for different fleet sizes (number of VFN = 6)	77

Figure. 4.1	Three-tier VFC system architecture highlighting Pub-Sub model implementation	84
Figure. 4.2	Flowchart of the proposed solution strategy indicating the role of each algorithm and their interdependence	87
Figure. 5.1	Performance evaluation setup and flow of simulations . . .	95
Figure. 5.2	Snapshot of the SUMO while running the simulation setup	96
Figure. 5.3	Snapshot of the FCD export file from SUMO during the simulation run	97
Figure. 5.4	Snapshot of the simulation setup in iFogSim2	97
Figure. 5.5	Latency values comparison of the proposed technique with previous competing works	102
Figure. 5.6	Variation in the number of VFNs selected by ACT-DW (dynamic resource adaptation based on the instantaneous workload demand)	103
Figure. 5.7	Energy consumption comparison	104
Figure. 5.8	Network bandwidth consumption comparison	105
Figure. 5.9	Maximum latency constraint satisfaction ratio	106

Symbols and Abbreviations

Symbols

Symbol	Description
t_{wv}^{tran}	Transmission time (for workload) between the vehicular node and VFN
t_{vf}^{tran}	Data transmission time (for workload) between VFN and FCN
t_{fc}^{tran}	Data transmission time (for workload) between FCN and CDC
t_{wv}^{prop}	Propagation time from vehicular node to VFN
t_{vf}^{prop}	Propagation time from VFN to FCN
t_{fc}^{prop}	Propagation time from FCN to CDC
t_v^{comp}	Computation time at VFN
t_f^{comp}	Computation time at FCN
t_c^{comp}	Computation time at CDC
n	Number of VN
N	Number of CVFN
θ	Normalized processor occupancy factor
δ	Latency
E	Energy consumption
B	Network bandwidth consumption
δ_{max}	Maximum permissible latency
F_{vr}	Optimal VFN for processing of the request r
γ	QoS satisfaction ratio
ψ	processing rate
α	workload size

Abbreviations

5GCIV	5 generation cooperative intelligent vehicular framework
AP	access point
AR	augmented reality
AVC	autonomous vehicular cloud
CAPEX	capital expenditure
CC	cloud computing
CDC	cloud data centre
CML	collaborative machine learning
CVFN	candidate vehicular fog node
DSRC	dedicated short range communication
FCD	floating car data
FCN	fog computing node
IoT	internet of things
IoV	internet of vehicle
ITS	intelligent transportation system
LAFF	location aware fog framework
LTE	long term evolution
MANET	mobile ad-hoc network
MBB	mobile broad band
MDE	modified differential equation
MEC	mobile edge computing
ML	machine learning
MOO	multi objective optimization
OBU	on-board unit
QoE	quality of experience
QoS	quality of service
RSU	road side unit
SDN	software defined networking
SLA	service level agreement
SUMO	simulation of urban mobility

V2I	vehicle to infrastructure
V2V	vehicle to vehicle
V2X	vehicle to everything
VANET	vehicular ad-hoc network
VCC	vehicular cloud computing
VFC	vehicular fog computing
VFN	vehicular fog node
VM	virtual machine
VR	virtual reality

Chapter 1

Introduction

Intelligent Transportation System (ITS) is the term used to represent use of information and communication technologies to improve safety, efficiency, and ease for users on the road (Alam, Ferreira, & Fonseca, 2016). An ITS is a ubiquitous network of heavily populated, energy and resource-constrained vehicular nodes. Each one of them capable of collecting and exchanging large volumes of heterogeneous data in real-time (Ning, Huang, & Wang, 2019). With the exponential growth of vehicles on road globally, it is expected to reach 2 billion number by 2030 (R. Hussain & Zeadally, 2019). The number of vehicles is increasing, and the on-board processing, communication, and storage capacity of the vehicles is increasing quickly. The ITS aims to improve the efficiency of road transportation systems, reduce road accidents, and help conserve the environment (Hou et al., 2016). Current generation vehicles have excess processing capability on-board, which can be advantageously used for ITS applications in smart cities (Tiwari & Kumar, 2015). With the upsurge in the processing capability of on-board devices, the demand for applications and services in the smart and connected vehicle domain is rising very fast. The Vehicular Ad-hoc Network (VANET) is a special case of the Mobile Ad-hoc Networks (MANET). In VANET, the vehicles with sufficient wireless communication facilities can form an ad-hoc network (X. Xu et al., 2020). In VANET vehicular devices can communicate from vehicle to vehicle (V2V) or from vehicle to infrastructure (V2I) (C. Zhu, Pastor, Xiao, & Ylajaaski, 2018). This chapter introduces the Cloud Computing, Fog Computing, and Vehicular Fog Computing in Sections 1.1, 1.2, and 1.3 respectively. Section 1.4 illustrates the motivation to carry out this research work, Section 1.5

describes the problem statement, and Section 1.6 presents the research objective.

Over past decade, the research in the area of VANET has witnessed a massive surge. Cloud computing, a centralized computing approach has been in use for mass computing, storage and integrating the data generated by connected /smart devices, because of the its advantages in terms of economy of scale towards deployment, operation and maintenance(Y. Zhang, Wang, & Wei, 2019). VANETs merger with cloud computing was proposed to form autonomous vehicular cloud (AVC). Further, Vehicular Cloud Computing (VCC) architecture was proposed so as to extend the capability of VANETs. VANETs later evolved to form the Internet of Vehicles (IoV)(J. Cheng et al., 2015). IoV services are an extension of the internet of things (IoT) in ITS domain. The IoV connects , humans within vehicles, intelligent systems on-board vehicles, and other cyber-physical systems in a smart city scenarios. The IoV integrates vehicles, on-board sensors, and mobile devices into a common network. It enables several new services to be rendered to vehicles and humans on-board and around vehicles. IoV extends VANETs scale, structure, and applications (J. Cheng et al., 2015).

Fog computing, an edge computing approach was proposed to be most suitable for many of the upcoming real time and latency sensitive applications, like emergency message transmission, autonomous driving, and smart traffic lighting system (Hou et al., 2016; B. Liu et al., 2021). Fog computing, is a supplementary computing technique to cloud computing, which is especially suited to real-time applications. It is characterised by : a) Ultra low latency and location awareness; b) Wide-spread distribution of computing resources (i.e servers); c) fast mobility; d) Arbitrarily large number of processing nodes; e) Wireless communication as major mode of communication; f) Streaming and real-time applications and; g) Heterogeneity of nodes. Fog provides storage, computation and services to users in a real-time, virtualized system. Fog computing enables processing of substantial information from sensors at the network edge, thereby conserving core network bandwidth and ensuring short latency experience for users(Eltoweissy, Olariu, & Younis, 2010; M. M. Hussain & Beg, 2021).

The intelligent transportation system (ITS) is a very promising application

area for fog computing (Grover Jyoti, Jain Ashish, 2018). ITS mainly relies on real time and low latency applications in connected vehicles environment. Fog computing is progressively finding increases role and applications in ITS environment. Despite exponential growth in research in fog computing, there are some important open issues and challenges related to its architecture, cost of implementation, security and efficiency for real time applications with quality of service (QoS) support. Fog computing approach permits QoS implementation while meeting application demand[18].]. Fog computing enables majority of the processing tasks to be processed at the edge of the core network, and the least possible data to be forwarded to the cloud. Therefore, fog computing is the most futuristic solution to ensure confidentiality (Khattak et al., 2019). Vehicular Ad-hoc Networks(VANET) are essential part of the Intelligent Transportation Systems(ITS) (P. Hu, Dhelim, Ning, & Qiu, 2017). VANET provides traffic efficiency, driving safety, and convenience by exchanging valuable information. They also provide support for various useful services like content-sharing applications (e.g., advertisements and entertainments) and information dissemination services (e.g., emergency operations for natural disaster and terrorist attack) (Chen et al., 2017; Contreras-Castillo, Zeadally, & Guerrero-Ibanez, 2018). Modern-day automated vehicles are connected vehicles, and they can provide universal access of data to vehicle users on the road. However, the ITS ecosystem with an increasing number of smart vehicles presents new issues such as universal, confidential, robust, scalable data exchange among vehicles, humans, and roadside units (RSUs) (Contreras-Castillo et al., 2018). With the increase in the number of vehicles and number of applications in ITS domain meeting demand of communication and computational capacity is becoming ever challenging. Some researchers presented ideas to use vehicles as computation infrastructure and termed it vehicular fog computing (VFC)(Hou et al., 2016).

The VFC uses a collaborative pool of user or near-user edge devices to complete the computation, based on efficient usage of individual processing resources of each vehicle (Hou et al., 2016; Khattak et al., 2019). It involves both parked and moving vehicles. Vehicles are parked for approximately 95% of the time in a day (Rahman, Yura Muhammad Iqbal, Newaz, Thien Wan, & Ahsan, 2019). Thus, parked vehicles become highly underutilized and thus, a

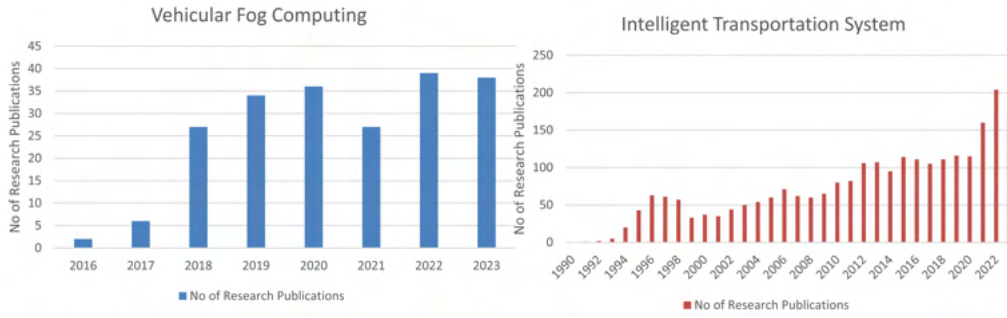


Figure 1.1 Evolution of Research in VFC and ITS

very convenient resource for VFC. By aggregating abundant resources of individual vehicles, the quality of services and applications can be greatly enhanced. Vehicular fog computing is a concept wherein the excess resources of vehicles are pooled to provide fog services in the ITS environment. Bottom layer in vehicular fog computing network represents vehicular fog, huge mass of resources collaborated by distributed vehicular nodes. Middle layer is conventional fog layer, performing tasks that cannot be handled by bottom layer. Further, top cloud layer is responsible for long term storage and high computational load and analysis tasks, that do not demand real-time response.

ITS has been very popular area of research over past decade. Whereas, research in the area of vehicular fog computing has attracted attention of researchers in last few years. Both these topics have seen steady growth in number of research publications, as depicted in Fig 1.1. As per google scholar search using key words intitle:”Vehicular Fog Computing”, in year 2016 total 2, in 2017 total 6, in 2018 total 27, in 2019 total 34, in 2020 total 36, in 2021 total 27, in 2022 total 38, and in 2023 a total of 37 research publications have been published. Similarly, google scholar search for key words intitle:”Intelligent Transportation System”, between 2010 and 2023 yields more than 100 research publications every year , on an average. Results depicted in Fig 1.1 indicate that VFC research is one of the fastest growing field in last few years, and ITS research has been a very hot research topic for almost a decade.

Despite advancement in communication networks in terms of 4G/ 5G etc in urban environment, availability and reliability of wireless communication is a challenge. There is also a huge surge in demand for computation resources in

urban vehicular applications, which requires large capital investment in terms of dedicated fog servers/ micro servers, road side units (RSU) etc. With modern generation of vehicles on board communication and computational capacity is increasing (K. Zhang, Peng, & Sun, 2020). Most of this capacity remains underutilized, that can be gainfully pooled for providing fog services to vehicular networks. VFC is a promising area of research, particularly with ever increasing on-board resources with modern vehicles.

With the steady rise in the on-board processing capacity of vehicles, requirement for novel vehicular services and applications is rising. The majority of such applications are latency-sensitive and have safety-related consequences. The latency is of critical importance for ensuring safety and quality of service (QoS) in ITS. However, achieving low latency, while meeting the workload demand from vehicular nodes in VFC is difficult because of dynamic changes in resource requirement and availability. Vehicular Fog Computing (VFC) makes computing power available much closer to the vehicles and end-user, thereby, reducing latency. The Internet of Vehicles (IoV) is a network for purpose of smart traffic management, dynamic vehicle information, and intelligent vehicle control. Thus, the IoV represents a typical application scenario of Internet of Things (IoT) technology namely, intelligent transportation systems (ITS). The use of IoV can also yield a better in-vehicle entertainment experience during the long journey. The IoV is an emerging field that has numerous applications such as autonomous vehicles, collision prevention, smart parking, smart traffic light systems, weather information, etc.

The IoV integrates connected vehicles in the IoT ecosystem, and this new technology amalgamation is leading to a new social revolution. The primary goal of the IoV is to enhance the safety of drivers and reduce congestion on the road. Accordingly, IoV applications are broadly categorized into two types namely, latency-sensitive (e.g., safety-critical) and tolerant-tolerant (e.g., road information, weather information, entertainment, etc). These two classes of applications have different requirements in terms of communication infrastructure and services architecture. The taxonomy of the IoV (i.e., vehicular network) applications is represented in Fig 1.2.

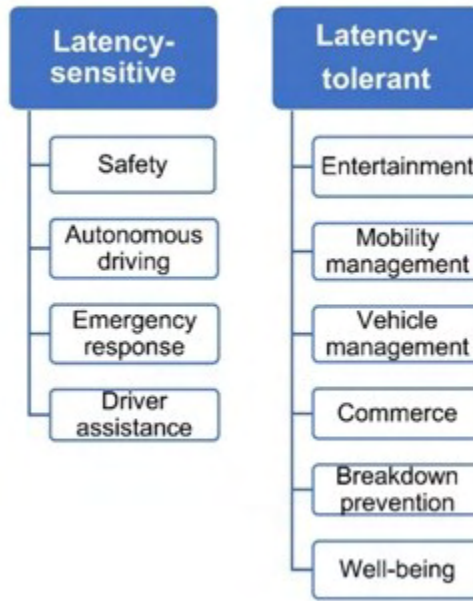


Figure 1.2 Taxonomy of vehicular network application

1.1 Cloud Computing

As IoV devices always have internet access to them, most applications for IoV are being developed with cloud computing services. Though the Cloud provides unlimited processing and storage, it has a few limitations. The most important of these is the high latency value. The high latency provided by Cloud computing is intolerable for safety-critical applications, that demand very low latency and support for stringent Quality of Service (QoS). A taxonomy computational technique is presented in Fig. 1.3.

1.2 Fog Computing

Computing technologies have evolved and are increasingly finding their application in IoV. In the case of latency-sensitive services processing, and storage should be performed near source devices so that latency and congestion of the network can be reduced. This led to a new research direction in the IoV

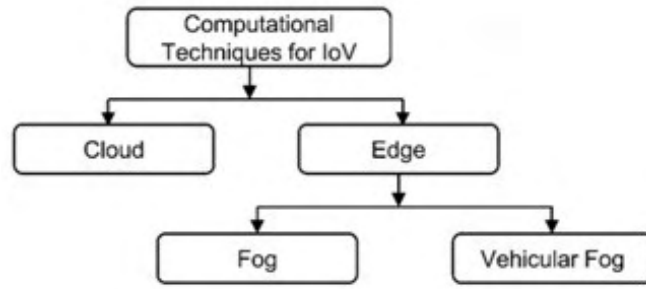


Figure 1.3 Taxonomy of computational techniques in IoV

ecosystem (i.e., to explore edge computing solutions). Fog computing, an edge computing solution, is more efficient as compared to the cloud in IoV, as it brings the computation power to the edge of the network, where most of the producers (i.e., sensors) and consumers (i.e., users or vehicles) of information reside.

Fog computing is definitely a promising solution for the IoV, it needs to be efficiently implemented to support QoS and enhance road safety. In the last few years, fog-enabled IoV has gained attention in the research community. Many researchers have proposed the architecture of vehicular fog network (VFN) and IoV integration with fog having three layers namely cloud, fog, and vehicular fog. There are still a lot of open research challenges to address concerning the architecture before the IoV becomes fully implementable.

The fog computing servers also called as fog nodes are geographically dispersed, in contrast to the fully centralized approach as in the cloud. Fog nodes should remain close to the network edge to efficiently serve the end users. In the IoV environment nodes are resource heterogeneous, and a fog node can be placed at any point on the network edge. The fog node can be any node (including moving vehicular nodes), and it may be dynamically selected, in contrast to traditional IoT fog nodes which are pre-defined and static. Considering mobility, resource heterogeneity of nodes, and safety-criticality of applications, selecting an optimal fog node in the IoV to meet low latency, low network congestion, and low energy consumption parameters is of paramount importance.

The IoV ecosystem has six main components to interact with each other and these are the vehicle's onboard units (OBU), person, personal device, communi-

cation network infrastructure, sensing device, and roadside units(RSU). The IoV interconnects humans with humans, vehicle intelligent systems, and numerous cyber-physical devices in the city environment by integrating sensors, vehicles, and other smart devices into a global network, thus allowing several applications and services to be rendered.

1.3 Vehicular Fog Computing

Although fog computing is suggested to be a capable answer for issues in IoV, it needs to be efficiently implemented to enhance road safety and support QoS. VFN collaborates to render a large chunk of computing resources on-demand (Hou et al., 2016) to the vehicles and users on them. VFN supplements vehicle-to-vehicle (V2V), vehicle-to-infrastructure (V2I), and vehicle-to-everything (V2X) communication, most commonly using Dedicated Short Range Radio (DSRC) and cellular (3G/ 4G/ LTE) radio networks (Shi, Lu, Zhang, & Yao, 2017; S. Liu, Xiang, & Punithan, 2018). VFNs provide very low latency that proves them most appropriate for latency-sensitive applications. The VFN resources are highly dynamic (Birhanie, Senouc, Messous, Arfaoui, & Kies, 2020; Meneguette, Boukerche, & Pimenta, 2019) thus optimal use of available resources on-demand, can be achieved using efficient resource management techniques. Researchers have proposed an architecture of VFC wherein, connected vehicles themselves also serve as fog nodes. The VFC demands no dedicated micro-server installation, instead, the moving vehicles collaborate to aggregate the surplus computational power with their OBUs to act as distributed fog servers.

1.3.1 The Benefits of Vehicular Fog Computing

Concept of VFC is promising as it aims at solving the following major constraints in real-time, latency-sensitive, and safety-critical applications in vehicular networks.

- By converting most of the multi-hop communication in fog-cloud architecture to single-hop communication, VFC minimizes transmission latency.

- It processes most of the requests from end users locally, thus reducing bandwidth consumption in the core network.
- The workload on the dedicated cloud or fog servers is also reduced which results in reduced Capital Expenditure (CAPEX) on infrastructure as well as substantial energy saving, paving the path for a green computing paradigm.

1.3.2 Dynamic Fog Nodes in Vehicular Fog Computing

The development in computing and communication technology, average computing capability of handheld devices and vehicle OBUs has been constantly increasing. Hence, user nodes in the vehicular network have surplus computing and storage capability which largely remains unutilized. simultaneously load on fog / cloud servers is rising owing to rising demand for a novel ITS applications. Such futuristic ITS services and applications demand massive processing and communication capability in fog devices to satisfy QoS and Quality of user Experience (QoE). The fog computing capacity can be augmented by pooling spare computing and communication capacity of user devices in the VFN landscape. This scheme, will reduce the cost to fog service providers and at the same time will generate revenue for the device owners, who can be paid for their device usage following the pay-as-you-use model by fog service providers.

The devices (i.e., vehicles) in a vehicular network are highly mobile. The mobility pattern of vehicles has huge variations and is not foreseeable. The mobility of devices in the VFN landscape poses two major challenges:-

- First, the workload (i.e., number of service requests) at any given location and at a given time changes dynamically, primarily because of the change in the number of end users requesting applications and services. User nodes' behavior and external factors such as weather, road conditions, road accidents, etc. are other factors responsible for such dynamic changes in workload.
- Second, the processing, storage, and communication capability available at any given location changes dynamically. This is due to the mobility and heterogeneity of fog devices.

Furthermore, the relationship between the dynamic services demand and dynamic resource availability is not predictable. Based on the above discussion, in the VFN landscape, a fog node must be dynamically assigned from available nodes to a particular task based on the instantaneous workload and fog resource availability at that location. While assigning the fog node for a application request, it is imperative to satisfy the QoS requirements of the application or service. Other desirable goals are the reduced energy consumption, network BW consumption, and the cost of fog services.

1.4 Motivation to Undertake This Research

The primary goal of ITS is to realize the vision of autonomous driving in future connected cities, which require real-time application with ultra-low latency. Because of the advancements in computing technology, future vehicles will have an excess computational capability, that can be proficiently utilized for providing fog computing services on demand resulting in VFC. VFC with dynamic and distributed architecture has a promising role to play in a smart city environment, especially in ITS. Efficient implementation of the VFC services in ITS has tremendous potential for improving road safety, reducing congestion on roads, and increasing comfort to road users. With the smart city concept being implemented in modern cities, the requirement of efficient ITS infrastructure, and scheduling algorithms to solve the challenges of latency, and energy efficiency are inevitable.

Two most important characteristics of vehicular network are first, dynamic resource demand and, second dynamic resource availability (Hou et al., 2016; Z. Zhou, Liao, Wang, Mumtaz, & Rodriguez, 2020) . The issues of minimizing latency, network BW, energy consumption, and QoS enhancement in VFC are being actively researched. (K. Zhang et al., 2020; Alharbi, Elgorashi, & Elmirghani, 2020; Mekki, Jmal, Chaari, Jabri, & Rachedi, 2020; Mahmud, Ramamohanarao, & Buyya, 2020; Zadobrischi & Damian, 2021) . The optimal positioning of vehicular fog nodes in VFC is critical to network efficiency and QoS compliance. The advent of Long Term Evolution(LTE)/ 5G cellular networks has enabled higher throughput, V2X communication links; thus,

the challenge of optimum scheduling of computational resources to process the latency-sensitive task, in vehicular networks becomes more pertinent (Alvi et al., 2022; Rahman et al., 2022). The following are key motivational points to undertaking this research work:

- ITS is rapidly expanding and a multi-disciplinary field of research involving various domains e.g., communication engineering, computer science, mathematics, etc. ITS application areas (e.g., autonomous driving, safety, entertainment, comfort) are expanding with the implementation of the smart city concept globally. Hence, the ITS domain has many open research challenges for researchers.
- Research in the fog computing domain is still gaining momentum. VFC is an important upcoming field of research. This topic has gained research momentum in last four to five years. Various research challenges in the VFC field are still undefined and unresolved.
- With the advent of Long-Term Evolution (LTE) / 5G cellular, artificial intelligence (AI), augmented reality (AR), and virtual reality (VR), the vision of autonomous driving is on the verge of becoming a reality. Ultra-low latency, distributed, and local decision-making provided by VFC is particularly significant for autonomous driving. Thus, latency minimization is one key open research challenge
- The problem of assigning real-time applications in VFC to optimal vehicular fog nodes for latency and energy optimization is critical for QoS enhancement for real-time safety-critical applications in VFC.

1.5 Problem Statement

Fig 1.4. depicts architecture of vehicular fog network. The vehicular fog network is a dynamic and resource-constrained environment. For efficient resource management, dynamic resource demand and availability of the vehicular fog resources in VFC need to be modeled accurately. Vehicular Fog Node (VFN) forms the identified processing resource in VFC ecosystem. The utilization of

resources in such an environment must satisfy the QoS requirements for ensuring safety on the road. The problem of modeling a dynamic vehicular fog node (VFN) has multiple conflicting objectives, for example, the latency requirement of any number of applications can be met by provisioning a huge chunk of dedicated resources, but that may result in unnecessary waste of resources most of the time when demand is less. VFC is a highly distributed system comprising multitude of the VFNs spread over the geographical locations of vehicles on the road. Therefore, the identification and selection of optimal VFN for processing of a latency-sensitive, safety-critical application becomes very critical problem for ensuring safety on the road. Thus, in VFC the assignment of application requests to most suitable available computing nodes (i.e. VFN) while meeting the QoS requirements is a challenging problem yet to be solved by academia and industry. In the context of the above statements following are important points identified for defining the problem for this research work.

- The VFC research is at a very nascent stage, and the architecture, characteristics, and application areas of VFC are in the forming phase.
- The real-time latency-sensitive and safety-critical applications in VFC are increasing at a fast pace, especially with the evolution of autonomous driving.
- The problem of minimization of latency and energy consumption has been studied by researchers in the past, considering the dynamic computational resources. However, the dynamic workload generation concurrently with dynamic computational resource availability is yet to be explored.
- The VFNs form a highly distributed and constrained computational resource in VFC, thus most optimal VFN must be assigned to the latency-sensitive and safety-critical vehicular applications dynamically (i.e. on a real-time basis) to achieve desired latency and QoS.
- The problem of dynamic VFN selection in VFC and assigning the latency-sensitive application requests to them for processing is critical to optimize the latency, QoS, and energy consumption.

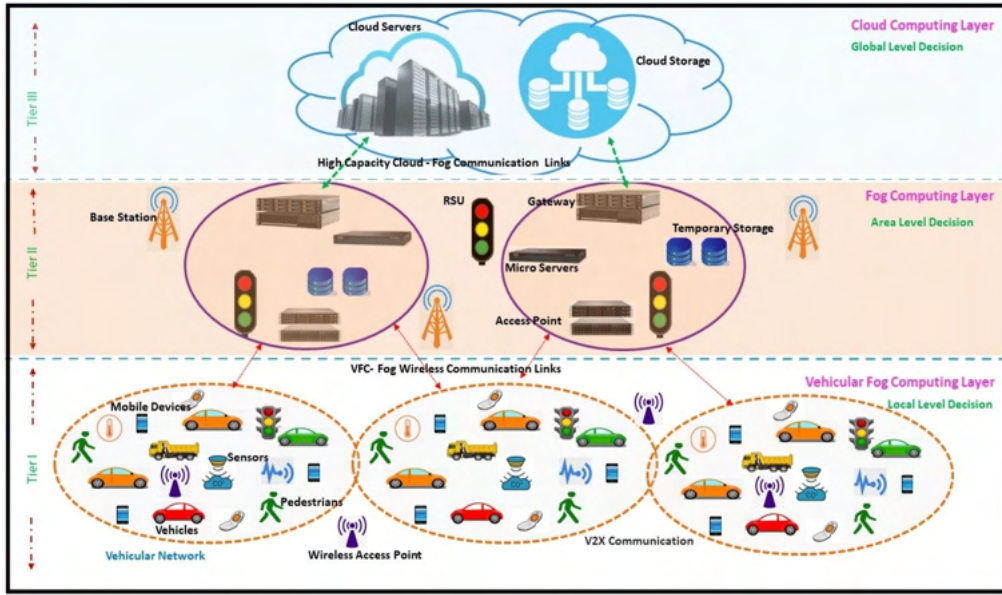


Figure 1.4 Architecture of Vehicular Fog Computing

1.6 Research Objectives

After analysis of the existing research in this domain and shortlisting the research problem statement, the following research objectives have been identified for this doctoral research. Primary objective of this research is to establish an efficient technique for dynamic fog node selection in vehicular fog networks to enhance the Quality of Service (QoS) and reduce energy consumption. The sub-objectives accompanying the main objective are as follows. The sub-objectives accompanying the main objective are as follows.

- To identify and model the optimal vehicular fog node characteristics.
- To develop and solve a multi-objective optimization (MOO) model for minimizing energy in an urban VFC scenario.
- To develop an energy-efficient algorithm(s) for QoS enhancement for the selection of dynamic vehicular fog nodes in urban VFC for intelligent transportation systems.

Chapter 2

Literature Review

2.1 Introduction

In this chapter, we discuss the details of the state-of-the-art research in the vehicular fog computing area. In Sections 2.1 to 2.4 literature is analyzed and based on analysis we present the characteristics, similarities, and dissimilarities of the cloud computing, and fog computing. Section 2.5 discusses architecture of vehicular fog computing systems. In Section 2.6 a summary of the detailed literature survey is presented. Section 2.7 presents analysis and comparison of the literature survey. Section 2.8 presents research challenges, and Section 2.9 summarizes the chapter.

2.2 Intelligent Transportation System

ITS is the use of information and communication technologies to enhance safety, efficiency, and comfort for road users (J. Zhang et al., 2011). An ITS is a network of thickly populated, energy and processing power-constrained devices, all of them capable of collecting and exchanging huge chunk of heterogeneous data in real-time basis (M. Hussain, Saad Alam, Sufyan Beg, & Akhtar, 2020). With the exponential growth of cars on the road globally, it is expected to surpass 2 billion numbers by 2030 (Gross, 2016). The number of vehicles is increasing and the processing, wireless communication and storage capabilities of these vehicles onboard are increasing significantly. ITS aims to make road transportation sys-

tems more efficient by improving road safety, and protecting the environment for future (Alam et al., 2016). Today's vehicular devices have excess hardware capability, which can be advantageously used to support ITS services in the smart city (Khattak et al., 2019). The surge in the hardware capability of on-board devices, the requirement for services in the connected vehicle domain is experiencing massive increase.

2.3 Evolution of Vehicular Network Research

VANET is a special case of the MANET. In VANET, the vehicles having wireless communication facilities can form an ad-hoc network (Darwish & Abu Bakar, 2018). In VANET vehicular devices can communicate from V2V or V2I (Kaiwartya et al., 2016). Over the past decade, research in VANET has witnessed a massive surge. Cloud computing, a centralized computing approach has been in use for mass computing, storage, and integrating the data generated by connected /smart devices, because of its advantages in terms of economy of scale towards deployment, operation, and maintenance (Satyanarayanan, 2017). VANET merger with cloud computing was proposed to form an autonomous vehicular cloud (AVC) (Eltoweissy et al., 2010). Further, Vehicular Cloud Computing (VCC) architecture was proposed to extend the capability of VANETs (R. Hussain, Son, Eun, Kim, & Oh, 2012). VANETs further evolved into IoV. IoV is an extension application of IoT in ITS (Sadiku, Tembely, & Musa, 2018). IoV connects people, sensors, vehicles, intelligent vehicular systems, and numerous cyber-physical systems in the city environment. By interconnecting vehicles, sensors, and mobile nodes. IoV enables several application services to be provided to connected vehicular nodes (Gasmi & Aliouat, 2019). IoV extends VANET's scale, structure, and applications (J. Cheng et al., 2015).

2.4 Fog Computing Characteristics

Fog computing, an edge computing approach was proposed to be most suitable for many of the upcoming real-time and latency-sensitive applications, like

emergency message transmission, autonomous driving, and smart traffic lighting system (Flavio Bonomi, Rodolfo Milito & Zhu, 2014; Bonomi, Milito, Zhu, & Addepalli, 2012). Fog computing is an alternative to computing paradigm of cloud computing, which is particularly suited to real-time applications. Important features of fog computing (Bonomi et al., 2012) are ultra-low latency and location awareness; geographical distribution of processing nodes; mobile nodes; huge number of vehicular nodes; wireless mode for seamless communication; demand of data streaming and real-time applications; variety of devices and services.

Fog provides storage, computation, and processing services in a real-time, virtualized system. Fog computing enables the processing of substantial information from sensors at the network edge, thereby conserving core network bandwidth and ensuring a short latency experience for users (J. Zhang et al., 2011; M. Hussain et al., 2020). The ITS is a very promising application area for fog computing (Grover Jyoti, Jain Ashish, 2018). ITS mainly relies on real-time and low-latency applications in connected vehicle environments. Fog computing is progressively finding increased roles and applications in ITS environment (P. Hu et al., 2017; Darwish & Abu Bakar, 2018). Despite exponential growth in research in this domain, there are few important research challenges related to its architecture, cost of implementation, security, and efficiency for real-time applications with QoS support. The fog computing approach permits QoS implementation while meeting application demand (Mahmud, Ramamohanarao, & Buyya, 2018). Fog computing allows a majority of the application request be materialized at the edge of the core network, and least application data be exchanged between user and the cloud. Therefore, fog computing is the most capable solution for ensuring confidentiality protection (Mendiboure, Chalouf, & Krief, 2019).

VANETs are an essential part of the ITS (P. Hu et al., 2017). VANET ensures efficiency, road safety against accidents, and ease to users by sharing useful data. They also provide support for numerous useful applications like streaming applications (e.g., advertisements and entertainment) and message dissemination services (e.g., emergency response and terrorist attacks) (Chen et al., 2017; Kai,

Cong, & Tao, 2016). Modern-day automated vehicles are connected vehicles, and they can provide universal reach to required data for vehicles and passengers on the road. The ITS ecosystem with an increasing number of intelligent/ connected vehicles poses challenges such as unified, confidential, reliable, scalable data interchange among vehicles, sensors, and RSUs (Contreras-Castillo et al., 2018). The surge in number of vehicles and applications in the ITS domain meeting demand for communication and computational capacity is becoming ever-challenging. Researchers proposed using vehicle OBUs as infrastructure for communication, and processing and termed it VFC (Hou et al., 2016).

2.5 Architecture and Applications of Vehicular Fog Computing

In (Alharbi et al., 2020), a virtual machine (VM) module placement technique has been proposed. The authors use the concept of multiple VM placement in a distributed, static fog environment to increase the QoS in comparison to the cloud. The authors bring out that multiple transmissions over the network increase the network power consumption, and offloading from cloud to fog reduces the overall power consumption. The parameters considered in the study are VM loading, traffic pattern/ density, and proximity of the fog node to the user. The vehicular nodes are considered for task generation but not for request processing in this work.

In (Alvi et al., 2022), the authors proposed a novel method of task scheduling by fog node in the ETEFN scheme, that schedules the execution of the task, utilizing the priority and left over time for completion within an RSU coverage. In this research work, authors have considered the processing capacity of fog nodes as a decision variable, but the authors have considered offloading the task to the cloud when the originating node is likely to go out of the RSU coverage range.

Authors in (Rahman et al., 2022) proposed a logical clustering system in VFC. Based on the trust value the VM assignment is done in the form of the most energy-efficient parked vehicle. The parameters to be satisfied are energy

consumption and communication interface capacity while selecting the vehicular fog for assigning a VM.

In (M. Hussain et al., 2020), the authors have studied the issue of reducing the latency and energy consumed as a combined value in VFC. The authors have presented a “task-offloading model based on probability and queueing theory” in VFC, fog, and cloud environments, and used a modified differential algorithm (MDE) approach for optimizing the proposed mathematical model.

In (Mahmud et al., 2018), authors have used fog-cloud architecture to develop an application placement policy for latency-sensitive applications, to implement QoS by reducing latency and optimizing for resource utilization. Authors have used access-delay, the delivery-time of services, and inter-nodal communication delay in placement of the application data in a distributed static fog environment. This research works to study the latency-sensitive applications in IoT scenarios (including vehicular applications) and aims at reducing latency and enhancing QoS, in fog-cloud architecture. However, the authors do not consider vehicles as fog computing nodes.

VFC uses a cooperative multitude of user edge devices to complete communication and processing, with the goal of achieving improved efficiency of specific communication and processing assets of specific individual vehicle (Hou et al., 2016; X. Xiao, Hou, Chen, Liu, & Li, 2019). It includes fast moving, slow moving and parked vehicles. Vehicles are parked approximately 95% of the time in a day (Rahman et al., 2019). Thus, parked vehicles become highly underutilized and thus, a very convenient resource for VFC. By pooling ample processing capacity from separate vehicles, the QoS of real-time applications can be greatly increased. VFC is a concept wherein the excess resources of vehicles are pooled to provide services in the ITS environment. The bottom layer in the vehicular fog computing network represents vehicular fog, the huge mass of resources collaborated by distributed vehicular nodes. The intermediate layer is conventional fog layer, performing tasks that cannot be handled by the bottom layer. Further, the top cloud layer caters for large storage and high computational load and analysis tasks, that do not demand real-time response.

Despite advancements in communication networks like 4G/ 5G etc in a city environment, the availability and dependability of wireless communication is a great challenge (Varghese & Tandur, 2014; Bennis, Debbah, & Poor, 2018). There is also a huge surge in demand for processing resources in a city scenario, which requires huge investments like dedicated fog servers/ micro servers, RSU, etc. With the modern generation of vehicles on board communication and computational capacity is increasing (Ieee, Ieee, Zhang, & Letaief, 2019). Most of this capacity remains underutilized, which can be gainfully pooled for providing fog services to vehicular networks.

Number of vehicles in cities is rising at an unprecedented rate globally. Additionally, the growth in communication and computational technology has led to a sudden surge in the number and variety of applications in the smart city environment. Some of these applications are computation intensive, whereas others demand a high level of context awareness and (or) ultra-low latency. Conventionally, cloud computing has been serving huge processing and storage-based applications, but not able to match the latency demand for real-time applications. Fog computing is supplementing the cloud computing for low-latency applications (P. Hu et al., 2017; Bonomi et al., 2012; Vaquero & Roderio-Merino, 2014; Baccarelli, Naranjo, Scarpiniti, Shojafar, & Abawajy, 2017; Barton, Goren, & Mahmoudi, 2018). Fog computing, makes use of network edge devices like routers, access points(AP), base stations (BS), micro fog servers, etc. to complement the cloud computing servers with low-latency solution (Hong, Tsai, & Hsu, 2016; Mayer, Graser, Gupta, Saurez, & Ramachandran, 2018).

The concept of using vehicles as infrastructure to augment the computational and communication capability of the vehicular network in an urban environment was proposed in (Hou et al., 2016) and the term VFC was coined for this. VFC is an extension of fog computing (M. Hussain et al., 2020; J. Huang, Qian, & Hu, 2020). VFC supplements fog computing, by collaborating additional computational resources near the end user premises. Thus, VFC provides a very attractive infrastructure for latency-sensitive tasks in ITS (P. Hu et al., 2017; J. Huang et al., 2020; Y. Xiao & Zhu, 2017; Tang, Wei, Zhu, Wang, & Jia, 2020;

Hoque & Hasan, 2019).

Many applications in the IoT and vehicular networks are latency-sensitive, which demands strict QoS. The parameters for QoS measurement in VFC are latency, network consumption, energy efficiency, and reliability (i.e. packet delivery ratio) (Mahmud et al., 2018; Shaheen et al., 2020). The QoS implementation has always been a challenge for service provisioning, especially so in a real-time application. The cloud computing (CC) solution, due to its centralized resource provisioning feature is not capable of QoS satisfaction in real-time applications.

In (Tang et al., 2020), authors have presented a scheduling strategy for time-sensitive and safety-critical applications in VFC. The three-tier computing architecture made up of VFC, static fog, and cloud has been used. Here, RSU has been used as computing gateways and storage units for data. However, in this work, the RSUs offload the tasks directly to the cloud server when resources are not available with vehicular fog. In (Shaheen et al., 2020), the authors have used the concept of a head node, that orchestrates, keeps track of other fog nodes, and proposed a "lightweight location-aware fog framework (LAFF)" to implement QoS satisfaction using the location-awareness algorithm. This work uses a two-tier architecture (i.e., fog and cloud) to implement QoS in IoT applications. The QoS improvement is measured by reduced latency, network BW usage, and resource consumption.

Researchers in (Yadav, Zhang, Kaiwartya, Song, & Yu, 2020), have studied the trade-off between the energy consumed and latency by application requests in VFC. Authors have proposed a dynamic computational offloading scheme (ECOS) for task offloading to vehicular fog, based on the overall energy optimization. The latency limits used for the study are in the range of 50 ms to 700 ms. The authors have used the concept of zone manager for the identification and tracking of the fog node for task offloading. In this work, QoS is defined by latency and energy consumption. However, the authors have not considered network bandwidth consumption as an optimization parameter, which is a key factor in reducing latency by avoiding core network congestion in the VFC sce-

nario.

An AI-based mobility management algorithm has been proposed in (Sodhro, Sodhro, Guizani, Pirbhulal, & Boukerche, 2020) for intra-vehicular networks. Further, the QoS implementation has been proposed based on mobility, trustworthiness, and packet drop ratio. In this work, focus is on better radio channel management for passenger QoS in vehicular fog networks. This research work does not include the static fog nodes during the investigation, which is an integral part of the three-tier VFC architecture. The VFC provides the most efficient alternative to the CC paradigm. For real-time IoT services, the CC is not very efficient, because of its higher transmission latency. A combined cloud-fog computing system is most suitable for collaborative computing to achieve latency minimization and energy efficiency. In (Hameed, ul Islam, Ahmad, & Munir, 2021), the authors have used the concept of a clustering, capacity-aligned load-balancing method for energy and performance-linked distributed processing for IoT applications. The dynamic clustering used in this work used vehicles' speed, direction, and position for clustering. The authors also propose a capacity-based intra and inter-cluster load-balancing technique for achieving a network-wide balanced energy consumption, minimum latency, and improved network BW utilization.

The demand for time-critical and data-intensive task computation is on the increase at network edge. In (C. Zhu et al., 2019), authors have proposed a quality-optimized task allocation technique with latency minimization. The research work considers the static fog, VFC and models the problem into a bi-objective optimization problem providing trade-offs between latency and service quality. The authors have not considered energy and network bandwidth consumption, the two very important parameters for optimization.

The authors in (Singh, Singh, Hedabou, & Kumar, 2023), have presented a computation resource allotment methodology in Software Defined Network (SDN) enabled fog computing system with "Collaborative Machine Learning (CML)". The CML system and resource allotment system is integrated with the proposed processing environment. In (Wei, Li, & Zhang, 2023), the authors

have presented a many-to-many offloading technique conceptualized on the resource trading for vehicles as a multi-tier offloading policy based on the trading consensus. Researchers in (Mattia, Pietrabissa, & Beraldi, 2023) proposed a load-balancing technique centered at the latency, aimed at leveling it across all the nodes in a decentralized manner. the goal of the authors in this work is that no one user will ever experience an inferior QoS than the other.

The authors in (G. Zhu et al., 2020), have proposed using machine learning (ML) along with communication networks for intelligent edge learning. A new term called “learning-driven communication” was coined. The overall idea of the edge learning concept is to utilize the communication network intelligently for utilizing the massive processing power at the network edge collaboratively, for minimizing latency. 5G-enabled software-defined networking (SDN) ITS system was proposed in (Din, Paul, & Rehman, 2019). However, in this work, the focus was on ensuring connectivity between vehicles, RSU, and, SDN controller and not on the application QoS. In (Rahman et al., 2022), the authors have studied the problem of energy efficiency vis-a-vis the VM⁴ migration in vehicular fog environments to support 5G applications. However, the authors assume two additional components namely the ”main broker (MB)”, and ”local broker (LB)” in the 5G core network in this research work. Though 5G-enabled VFC networks are a promising solution for addressing latency and bandwidth issues in ITS, they have one very important challenge, privacy protection.

In (Boualouache et al., 2021), the authors have studied the issue of confidentiality preserving and presented a blockchain-based approach to address it. In (X. Cheng, Chen, Zhang, & Yang, 2017), the authors have brought out the key features and challenges associated with the 5G co-operative intelligent vehicular (5GCIV) framework. The research work highlights code-v as a fusion of communication and intelligence, thus bringing forth the important challenge of the 5G framework to incorporate the features at the core of the network for supporting ITS requirements. The authors in (M. M. Hussain & Beg, 2021) bring out many challenges associated with the current implementation of the 5G-enabled VFC approaches and propose CODE-V, a multi-hop task offloading in VFC using DSRC and LTE for V2X communication.

The ITS with connected vehicles aims at reducing traffic accidents, reducing traffic congestion, and providing convenience to road users (Alam et al., 2016). Many applications are being developed for connected vehicles to meet these requirements. It is expected that 150 million cars will be connected cars of one or another type by the year 2020 (Ning et al., 2019). With the current trend of driver-less i.e., autonomous vehicle testing in the leading cities of the world, vehicular networks have become very important for safety on the road. Present-day latency-sensitive ITS applications cannot tolerate delays provided in transit and data analysis by cloud servers (R. Hussain & Zeadally, 2019). This is a serious issue with safety-critical applications like autonomous driving. VFC integrates fog computing and VANET, which aims at the collaborative pooling of massive computational, and communication resources from the vehicles (Hou et al., 2016; Tiwari & Kumar, 2015). Many new application areas are forthcoming in connected cars like entertainment, e-commerce, advertising, route planning, and automated car parking system (Ning et al., 2019; Hou et al., 2016; X. Xu et al., 2020; C. Zhu et al., 2018; Y. Zhang et al., 2019; Tiwari, Sharma, Upadhyay, Sachan, & Sharma, 2019).

2.6 Summary of Literature Review

A summary of the important research papers in VFC domain studied during systematic literature review is presented in Table 2.1. The summary includes the techniques utilized, parameters optimized, and major contributions of the important research work.

Table 2.1 Summary of the systematic literature review

Ref, year	Technique(s) utilized	Performance metrics details	Parameters optimized	Important contribution(s)

(Alharbi et al., 2020), 2020	Offloading from cloud to fog	reduced power consumption compared to cloud	VM loading, latency	VM-module placement strategy
(Alvi et al., 2022), 2022	Task priority, node's processing capacity	reduced total energy consumption in FN	Energy consumption, remaining time in RSU coverage	Energy-Efficient task-scheduling and execution
(Rahman et al., 2022), 2022	Logical clustering based on trust value	most energy efficient parked vehicle as fog node	Energy consumption	Energy efficient VM assignment from parked vehicles
(M. Hus-sain et al., 2020), 2020	Probability and queuing theory	minimum combined value of latency and energy consumption	Latency, energy consumption	Task offloading model
(Mahmud et al., 2018), 2018	Distributed static fog nodes	reducing latency and enhanced QoS	Service access delay, internal communication delay, and resource utilization	Application placement policy for latency-sensitive applications
(Tang et al., 2020), 2020	RSU as computing gateway	QoS enhancement for real-time applications	Latency	Task scheduling strategy
(Yadav et al., 2020), 2020	Computation offloading scheme	QoS (Latency and energy consumption) reduction	Latency, overall energy consumption	Energy-latency trade-off

(Sodhro et al., 2020), 2020	AI-based radio channel management	QoS enhancement	Mobility, packet loss ratio	Radio interference-free, trustworthy mobility system
(Hameed et al., 2021), 2021	Clustered, capacity-linked load balancing	latency minimization, bandwidth reduction, energy balance	Latency, energy, and network consumption	Dynamic clustering, inter/intra-cluster load balancing
(C. Zhu et al., 2019), 2019	Latency service-quality trade-off	latency and service quality trade-off	Latency, service quality	Latency and service quality based task allotment algorithm
(Singh et al., 2023), 2023	SDN based collaborative machine learning (CML)	minimum resource consumption	Latency, energy, network usage	CML in SDN environment for enhancing QoS
(Wei et al., 2023), 2023	Latency service-quality trade-off	resource consumption reduction	Latency, resource utilization	Many-to-many task scheduling and vehicular trading in VFC
(Mattia et al., 2023), 2023	Differential model using adaptive heuristics	latency and QoS	Service latency	Load balancing aimed at leveling low-latency across all the nodes
(G. Zhu et al., 2020), 2020	ML-based intelligent edge learning	minimizing latency	Latency, resource utilization	Learning-driven communication

Many of the research works discussed above have proposed to optimize QoS for real-time applications. Researchers have used different approaches, and techniques for achieving the desired results. However, none of the existing works have categorized the tasks into latency-sensitive and latency-tolerant. The simulation in all previous works are based on the generic VFC scenario. All the existing works have taken dynamic workload into consideration but, none have considered the dynamic resource availability. The proposed work uses hybrid scheduling and dynamic network segmentation techniques to enhance QoS. As compared to previous research works, the proposed research work distinguishes itself based on the following unique contributions. Firstly, the requests are classified into latency-tolerant, and latency-sensitive with each latency-sensitive task having a well-defined maximum permissible latency. The latency-sensitive request must be processed within the defined maximum permissible latency limits, otherwise, the results may lose their utility. Secondly, in the proposed research work the task latency and total energy consumed is optimized while satisfying the maximum permissible latency limit for the request. Thirdly, the novel algorithm for an optimal number of CVFNs ensures the minimum cost to the fog service provider. Lastly, this research work uses a multitude of parameters (i.e., speed, direction and location of fog nodes, processing capability, instantaneous processor load, maximum latency limit of request, speed, location, the direction of the source node, communication capacity and, type of request) for selection of optimal VFN.

2.7 Analysis of the Comprehensive Literature Review

From the comprehensive literature review and analysis it is brought out that the computational techniques for IoV applications have evolved over a while. Cloud computing has been reigning in the remote computation domain for long. With the advent of high computational capability of devices and high-speed wireless networks, idea of network edge computing evolved. Further research in edge computing domain led to two major subdomains of edge computing namely fog computing and VFC.

Cloud computing has been dominant till the year 2000. Cloud computing is most suitable for applications that need a massive amount of computing power but has limitations in implementation requiring very low-latency. With the advent of real-time applications edge computing and fog computing were proposed by researchers around the year 2000. Smart vehicles enhanced the capability of the vehicles in ITS, thus there was a need for a more efficient and latency/QoS-optimized computing paradigm. The VFC dimension has been researched since the year 2020. Approximate timelines for the evolution of techniques are depicted in Fig 2.1. The hierarchical architecture of VFC is depicted in Fig 2.2. Three-tier VFC architecture consists of, VFC, fog-computing, and cloud layers.

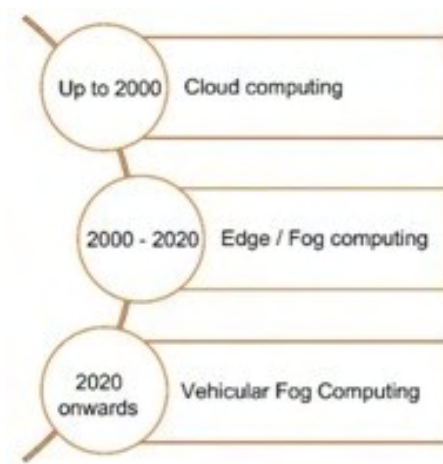


Figure 2.1 Evolution of computing technology

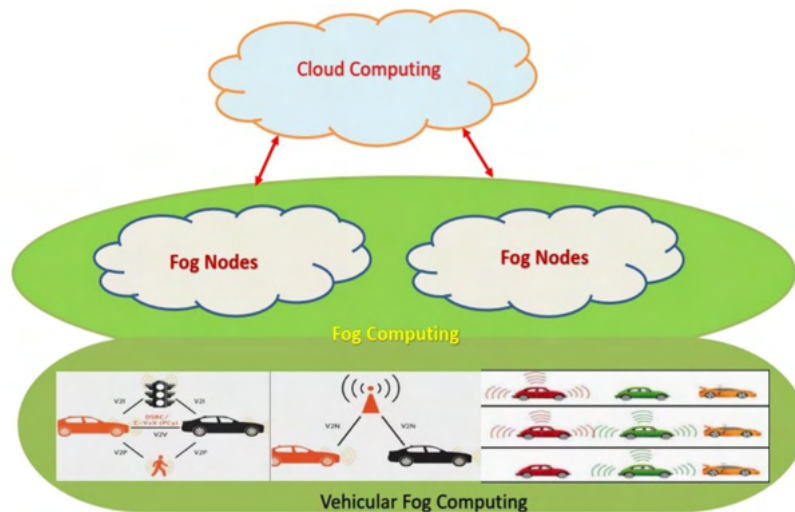


Figure 2.2 Hierarchical architecture of VFC

2.7.1 Characteristics of Vehicular Fog Computing

From the discussion and analysis of literature, we can deduce the following important points about VFC. VFC is an evolution of, and complements fog computing. It encompasses user devices and vehicles as VFNs, conventional fog computing nodes, and cloud servers typically arranged in hierarchical order. VFC provides an inexpensive and dynamic pool of computation and communication resources closest to the end user. Important characteristics of VFC are as follows:-

Low latency and real-time processing

VFC nodes are special kinds of fog nodes, that assume roles of the end user as well as fog nodes. The end users are connected to these vehicular fog nodes in a single hop or few hops (Khan, Abolhasan, & Ni, 2018). These vehicular fog nodes are connected to other nearby vehicular fog nodes by a short distance, V2V Communication (Hou et al., 2016). Such V2V communication in dense urban traffic conditions provides much better coverage as compared to cellular, Wi-Max, Wi-Fi, etc, with minimum coverage gaps. The reliability, throughput, and availability of these communication links are higher. VFC is therefore best suited for latency-sensitive, real-time data processing.

Location awareness

Fog computing, as opposed to cloud, delivers location-based services. VFC, an evolution of conventional fog also supplements location awareness (Tang et al., 2020; Khaliq, Chughtai, Shahwani, Qayyum, & Pannek, 2019). Majority of the vehicular network applications and services necessitate location awareness (Hoque & Hasan, 2019). VFC permits absolute location awareness because most of the requests are served by the VFC nodes placed very near the end user location.

Optimum network bandwidth utilization

Fog computing helps efficient use of available network bandwidth by limiting the traffic flow in the core network, which connects to cloud data centers (OpenFog Consortium Architecture Working Group, 2017). Similarly, in vehicular fog

computing, most of the requests from end users are served in the vehicular network itself. Only those services and tasks that are very high compute and (or) storage intensive will be handed over to fog or cloud for computing. Therefore, the core network bandwidth usage and hence, congestion is minimal (J. Huang et al., 2020; L. Liu, Chen, Pei, Maharjan, & Zhang, 2019). This results in the optimum utilization of the core network infrastructure.

Support for mobility

The very important characteristic of VFC that distinguishes it from other computing systems is mobility support (Hou et al., 2016; Hoque & Hasan, 2019; Ran & Bai, 2020). In VFC static and slow moving vehicles are used as processing nodes (Hoque & Hasan, 2019). The parked vehicles on the one hand make a good choice (both computational and communication) for hub nodes connecting other moving vehicles. Moving vehicles provide augmented computational capacity to support burst applications and services. Slow-moving and parked vehicles provide excellent, long time computational resources. VFC uses best characteristics of slow-moving and parked vehicles (Hou et al., 2016).

Dense geographic distribution

In ITS, VFC nodes are situated in OBUs of the vehicles. Thus, intrinsically VFNs are densely populated in the urban settings.(Ran & Bai, 2020). This feature of VFC differentiates it from static fog. Vehicular fog networks are particularly suited for latency-sensitive, real-time applications in urban traffic conditions as part of ITS(Tang et al., 2020).

Dynamic resource availability

Because of the mobility of vehicles, VFC has dynamically changing workloads and resource availability in both time and location. Two cases of static VFC (i.e. parked vehicles as fog nodes) and dynamic VFC (i.e. moving vehicles as fog nodes) are possible (Mekki, Jabri, Rachedi, & Ben Jemaa, 2018). Dynamic availability of resources has benefits i.e., it reduces cost by avoiding excess installation and under-utilization of these computing assets. The limitation is the need for highly sophisticated co-ordination system and hand-off mechanism

(Ran & Bai, 2020).

No installation cost

VFC uses co-operative vehicular nodes, which collaborate to pool huge computation and communication resources. Thus, no dedicated fog infrastructure is required to be installed. Moreover, the availability of such resources is proportional to number of vehicular nodes in a certain region at any given point in time. The inference drawn from this is that the more the number of vehicles N_v , the more the demand for resources D_r , and the more computing resources A_r are available. Availability of processing resources is driven by an increase in demand for resources. $N_v \propto D_r \propto A_r$

Decentralized data processing

One of the most important characteristics of the VFC is that resources are decentralized (Ning et al., 2019; Sookhak et al., 2017). It implies that sufficient redundancy and reliability of computing and communication resources are available. In VFC networks most dominant communication mode is V2V, which results in fewer network bottlenecks in comparison to infrastructure communication modes like cellular, Wi-Fi, WiMax, etc. Decentralized computing resources also make such networks agile and highly rugged against node failure.

Heterogeneity of resources and applications

VFC networks are made up of pooled resources from the available OBUs. Resource, as well as the application requirements in VFC, are heterogeneous (Ning et al., 2019). QoS will depend on, how judiciously, these resources are allocated to demanding applications and services. Allocating these heterogeneous resources, optimally, to heterogeneous requirements is a challenge (Hoque & Hasan, 2019).

Security and privacy protection

This is a major concerns in the case of VFC. Firstly, the connectivity graph of the VFC nodes keeps changing because of the movement of vehicles. Secondly, authentication and authorization become a difficult due to the heterogeneous and

continuously varying number of nodes (R. Hussain & Zeadally, 2019; Hoque & Hasan, 2019; Bennis et al., 2016). Security and privacy concerns in VFC are like those of VANETs and fog networks.

2.7.2 Comparison of Vehicular Fog Computing and Conventional Fog Computing

VFC seems identical to fog computing, it has a few characteristics that make it unique and different from conventional fog computing. In VFC, fog nodes or servers are in much close proximity to user. The VFC nodes are OBUs of vehicles. Some of the end-user devices themselves serve as vehicular fog computing nodes. End users communicate with VFC nodes in one or a few hops, predominantly, using V2V communication (Ning et al., 2019; Hou et al., 2016; Hoque & Hasan, 2019). As opposed to VFC, fog computing nodes are installed in RSU, Access Points (AP), Mobile Base Stations, routers, and micro data centers (P. Hu et al., 2017; OpenFog Consortium Architecture Working Group, 2017; Marín-Tordera et al., 2017). In conventional fog computing architecture fog layer(s) exist between user nodes and the cloud. Whereas, in the case of VFC architecture, fog servers are part of the end-user vehicular network. VFC nodes have a more dense geographic distribution as compared to conventional fog nodes. Fig 2.3 represents the interrelationship among major characteristics of VFC, fog computing, and the cloud.

Conventional fog nodes support low or no mobility, in contrast, VFC nodes support mobility. The clustering nodes is vital for best processing asset usage and QoS support (Hou et al., 2016; X. Xu et al., 2020; Fan, Shi, Chen, & Zou, 2012; Anjum & Khan, 2019). Considerable processing capability can be aggregated in VFC in comparison to static fog, without incurring any installation cost (X. Xiao et al., 2019; Y. Xiao & Zhu, 2017; Kui, Sun, Zhang, & Li, 2018; Sami & Mourad, 2020). More number of VFC nodes in comparison to static fog nodes indicates geographically distributed computational resources. Such decentralized distribution of resources is particularly desirable in applications with strict QoS demand, and location awareness. Examples of such applications are traffic management/congestion control, accident information distribution,

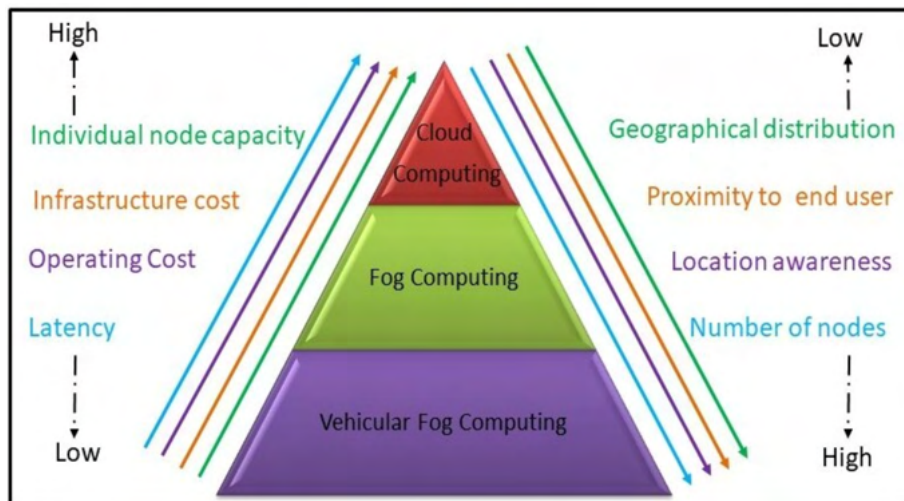


Figure 2.3 Comparison of the characteristics of vehicular fog, fog, and cloud computing characteristics

content sharing, advertising, etc in an urban environment. The differences between conventional fog and VFC network are presented in Table 2.2.

Table 2.2 Differences between conventional fog and VFC

Parameter	Fog Computing	Vehicular Fog Computing	Insight
Proximity to the end user	Medium	High	The highest order of proximity to end users will ensure the least latency.
Geographical distribution density	Medium	High	High geographic distribution density means, higher order of reliability against node failure and more precise location information.
Decision making	Distributed / local	Local	Local (i.e., vehicle level) decision-making results in the fastest decisions as compared to conventional fog.
Mobility support	No / limited	Yes	The fast speed of vehicles may result in certain tasks not being completed by one VFC node. Mobility support will ensure seamless task handover.
Clustering of nodes	No	Yes	Most of the VFC nodes are on the move. Clustering of nodes based on mobility pattern etc. will result in efficient utilization of the VFC network.
The capacity of individual nodes	Medium	Low	Individual nodes pool their unutilized resources, which are not dedicated fog resources. Individual node capacity is usually low as compared to conventional fog nodes.

Number of nodes	Medium / low	High	Because of high vehicular density in ITS, a more of VFC nodes are available. Whereas conventional fog nodes are dedicated fixed and limited resources.
Overall capacity	Medium	High	The overall capacity of VFC is much higher in comparison to the fog system, because of a much larger number of nodes.
Computational capacity availability	Static	Dynamic	VFC networks are formed dynamically, on the fly. The capacity of the network keeps changing as per resource availability, unlike conventional fog where it is fixed.
Fog node location	Routers, AP, RSU, Base Station, etc	Vehicle OBU	The location of VFC nodes is the OBU of vehicles, as opposed to the static locations of conventional fog nodes.
Communication modes	Cellular, Wi-Fi, Wi-Max, and wired	V2V, V2I, V2P	Modern OBUs of vehicles are equipped with multiple interfaces for V2X communication.
Location awareness	Medium	High	The nodes in VFC are more dispersed geographically. Thus, precise location information is available.
Cost of installation	Medium /High	Nil	No dedicated fog nodes are installed in VFC. Hence, a nil installation cost is required.

The VFC is a special case of (conventional) fog. Therefore, vehicular fog computing also shares many common features with conventional fog. The Venn

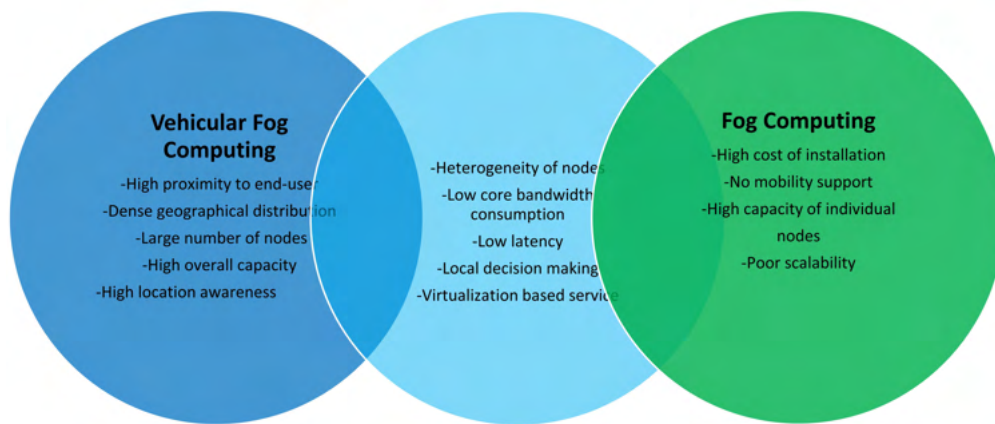


Figure 2.4 Comparison of the characteristics of Vehicular Fog and Fog

diagram in Fig 2.4 depicts the comparison of their characteristics. Table 2.3 presents the similarities between fog and VFC.

Table 2.3 Similarities between conventional fog and VFC

Parameter	Fog Computing	Vehicular Fog Computing	Insight
Architecture	Hierarchical, decentralized, distributed	Hierarchical, decentralized, distributed	VFC is an extension of conventional fog. It has the same basic architectural features as that conventional fog.
Latency	Low	Low	Both are suitable for low-latency, real-time applications.
Core bandwidth consumption	Low	Low	Both these techniques complete most of the tasks without sending them to cloud servers via the core network. Thus, core bandwidth consumption is minimum.

Heterogeneity of nodes	Yes	Yes	Types of nodes are not standardized. Thus, managing heterogeneous nodes is a common characteristic of both these techniques.
Scalability	High / medium	High	Both VFC and conventional fog networks are easy to scale up/down. Easy to add or remove nodes.
Virtualization based service	Yes	Yes	Both these computing paradigms use the concept of virtualization of hardware resources. This ensures optimal use of available resources.

2.7.3 Application and Services in VFC

VFC is a form of the fog computing created on ad-hoc by using the surplus computational assets of nearby vehicular nodes (Sami & Mourad, 2020). Worthy and volunteer nodes subcontract unutilized OBU processing power to VFC network. Clients in VFC can use this pooled processing power for numerous computational roles (Hoque & Hasan, 2019; Liao et al., 2020; Iqbal, Malik, Rahman, & Noor, 2020). Therefore, a VFC forms a vast chunk of resources, pooled by vehicular OBUs. Vehicular fog network resources can be employed for providing the following types of services on similar lines as the fog resources (Hoque & Hasan, 2019): -

- Information as a service (IaaS)
- Entertainment as a service (EaaS)
- Storage as a service (SaaS)
- Computation as a service (CaaS)

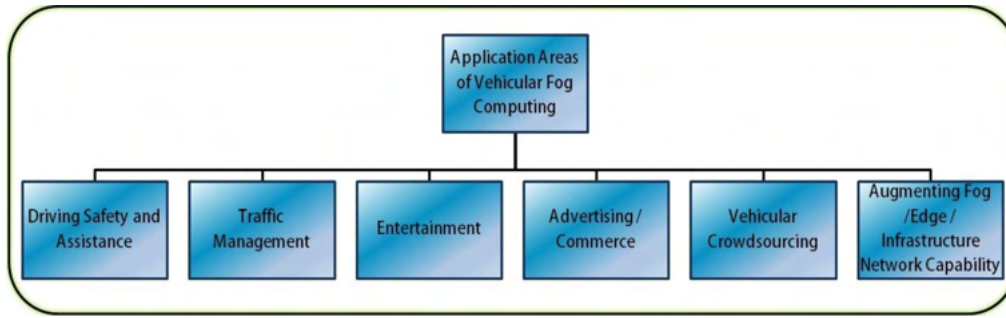


Figure 2.5 Categories of VFC applications

2.7.4 Applications of VFC

VFC applications are increasing very fast. This is particularly so because vehicles are witnessing a sharp increase in the computational and communication capability of OBUs. Researchers have proposed many new applications of VFC in the smart city environment. Based on our survey of existing literature and analysis of the future scope of VFC applications, we have categorized VFC applications into six broad areas as depicted in Fig 2.5. Table 2.4 represents the summary of potential applications in VFC. Potential applications of vehicular fog computing are summarized in Table 2.4 (Ning et al., 2019; Hou et al., 2016; X. Xu et al., 2020; C. Zhu et al., 2018; Y. Zhang et al., 2019; Kui et al., 2018; Vergis, Komianos, Tsoumanis, Tsipis, & Oikonomou, 2020; S. Zhou, Sun, Jiang, & Niu, 2019; Wu et al., 2020; Fu, Kang, Zhang, Yu, & Wu, 2020; M. Li, Zhu, Zhang, Du, & Guizani, 2018; Z. Wang, Zhong, & Ni, 2018).

Table 2.4 Potential applications of vehicular fog computing

Broad application domain	Specific applications	Description
Driving safety and assistance	Assisted driving	The ability for a car to sense and anticipate potential dangers long before the driver senses it
Driving safety and assistance	Accident reporting	Automatic accident detecting and reporting system
Driving safety and assistance	Vehicle theft reporting	Disseminating vehicle theft information

Driving safety and assistance	Smart emergency response system	Quick and automatic road emergency response system
Driving safety and assistance	Autonomous driving	Self-driving and driver-less, connected vehicles in smart city
Driving safety and assistance	Decision support system for road safety	A data-driven MIS to improve road safety
Traffic Management	Smart /Adaptive traffic light	Adaptive traffic light system to intelligently route vehicular and pedestrian traffic
Traffic Management	Traffic congestion control	Traffic congestion control mechanisms based on vehicular load, road conditions, accidents, etc.
Traffic Management	Navigation assistance	Smart navigation and route planning for reduced congestion and better economy
Traffic Management	Parking assistance	Assistance for accurate and efficient parking system on congested roads and market
Traffic Management	Smart Parking system	Integrated, real-time system for managing limited parking space efficiently and conveniently
Traffic Management	Car-pooling assistance	Pooling of cars by users of the same route to reduce congestion on the road
Traffic Management	Taxi service planning	Taxi service planning and positioning as per instantaneous passenger load
Entertainment	Video streaming /file sharing	Content sharing among people traveling in the same area to reduce delay and network load
Entertainment	AR/VR-based gaming	Augmented /virtual reality-based games, requiring high-performance computation and communication

Entertainment	Internet connection sharing	Sharing internet connectivity to devices in the local area via V2V connection
Advertising /Commerce	Location-aware advertising	Advertising commodities/ services that are of interest to people traveling in a specific area
Advertising /Commerce	Targeted advertising	Advertising to a people or groups, based on their specific interest
Advertising /Commerce	Tourism information dissemination	Targeted, information dissemination to tourists/interest groups
Vehicular crowd sourcing	Video surveillance	Inputs from video cameras installed along the roadside for surveillance of the area
Vehicular crowd sourcing	Road condition monitoring	Inputs from sensors on vehicles traveling on road to monitor road conditions in that area
Vehicular crowd sourcing	Route information sharing	On-board sensors on vehicles contribute inputs to provide accurate route information
Vehicular crowd sourcing	Accident prediction and analysis	Sensors on-board vehicles and on road contribute data continuously towards analysis. These results are used to predict and prevent accidents
Vehicular crowd sourcing	Accurate weather information collection	On-board vehicular sensors, which have dense geographic distribution, can augment stationary meteorological stations for providing more accurate weather information

Vehicular crowd sourcing	Leasing of underutilized resources of vehicles to neighboring vehicles	Co-operative computing environment, wherein, underutilized resources of vehicles are leased to neighboring vehicles on-demand basis
Augmenting Fog /Edge / Infrastructure network capability	Task offloading from fog/edge networks to vehicular fog network	Compute resources at the edge or conventional fog network can be supplemented by offloading tasks to nearby VFC networks with sufficient resources available
Augmenting Fog /Edge / Infrastructure network capability	Task offloading from cellular network /MEC to vehicular fog network	Compute tasks from nearby infrastructure networks can also be offloaded to the VFC network with sufficient resources, via V2I communication
Augmenting Fog /Edge / Infrastructure network capability	Detecting and bridging mobile broadband coverage gaps from Wi-Fi/ Wi-Max/ 3G/ 4G/ 5G to end users	VFC network can also augment internet connectivity in the local area by detecting mobile broadband coverage holes and extending broadband connectivity (i.e., Wi-Fi/Wi-Max 3G/ 4G/ 5G) to end users/sensors

2.7.5 New Applications of VFC Identified

Applications of the VFC will increase safety, efficiency, comfort, and revenue in ITS in the future smart city. Few of the applications have already been explored by researchers. However, based on our study and analysis, the following new application(s) /areas have been presented in this research work:

Driving Safety and Assistance

- Decision support system for road transport planning as an MIS tool for improvement in road infrastructure and movement planning.
- Vehicle theft reporting system for quick response and recovery of the stolen vehicle.

Entertainment

- Internet connection sharing among road users using V2X communication.

Advertising/commerce

- Location-aware and targeted advertising for effective reach to the target audience.
- Tourism information dissemination for tourists traveling/destined to a specific area.

Vehicular Crowd-sourcing

- Real-time road condition monitoring system using vehicle crowd-sourcing as a source of reliable input.
- Real-time route information sharing using vehicle crowd-sourcing for efficient and economic route planning.
- Accurate weather information collection system using vehicle crowd-sourcing.

Augmenting fog/edge/infrastructure network capability

- Task offloading from cellular network /MEC (mobile edge computing) to vehicular fog network to reduce the load on edge network, and at the same time optimally utilizing nearby underutilized VFC network.
- Detecting and bridging mobile broadband coverage gaps from Wi-Fi/ Wi-Max/ 3G/ 4G/ 5G to end users by using the V2X communication of the VFC network to enhance QoS for Mobile Broad Band (MBB) service.

2.7.6 A Simple VFC Application Use-case Illustrated

Bridging coverage gap of mobile broadband: A VFC application use-case

Smart cities / urban areas have seamless wireless broadband coverage for mobile users/devices utilizing cellular services, WiFi, WiMax, etc. A common phenomenon in MBB is packet loss resulting in low QoS. Wireless packet loss is due to coverage gaps/ dark spots (Baltrunas, Elmokashfi, Kvalbein, & Alay, 2016). Sensing, filling the network black spots in wireless MBB is a vital research problem (Ali et al., 2020). Vehicular fog networks can be of use in such a scenario to detect and fill the coverage gaps. Fig 2.6 illustrates one example use-case for a vehicular fog network to fill the coverage gap by extending the cellular network coverage to gaps.

Vehicular fog computing networks have very dense geographic distribution. On-board vehicular fog nodes have multiple communication interfaces with them. Nodes connect to mobile broadband stations via V2I communication. Vehicular fog nodes within a VFC network communicate among themselves using V2V communication (single hop or few hops). MBB coverage could be provided utilizing cellular (i.e., 3G/ 4G /5G) base stations or Wi-Fi or Wi-Max access points. On-board sensors on vehicles can detect the breaks or holes in the coverage of MBB. Whenever a coverage gap is detected, adjacent vehicular fog nodes (i.e., under MBB coverage) will relay the broadband services to nodes not covered by MBB, using V2V communication. The vehicular fog nodes can also act as AP or hot-spot for the users/ devices outside the coverage zone. Hence, in the above-mentioned scenario, the VFC network can supplement MBB coverage on-demand basis to enhance QoS.

2.7.7 Classification of Vehicular Network Applications

Based on the comprehensive literature survey and its analysis, in this research work vehicular networking applications have been categorized into two major classes as depicted in Fig 2.7. This classification relates to most important characteristics of VFC application i.e., latency. Latency-sensitive applications are safety-critical e.g., turn radius indicator, accident-avoidance system, autonomous

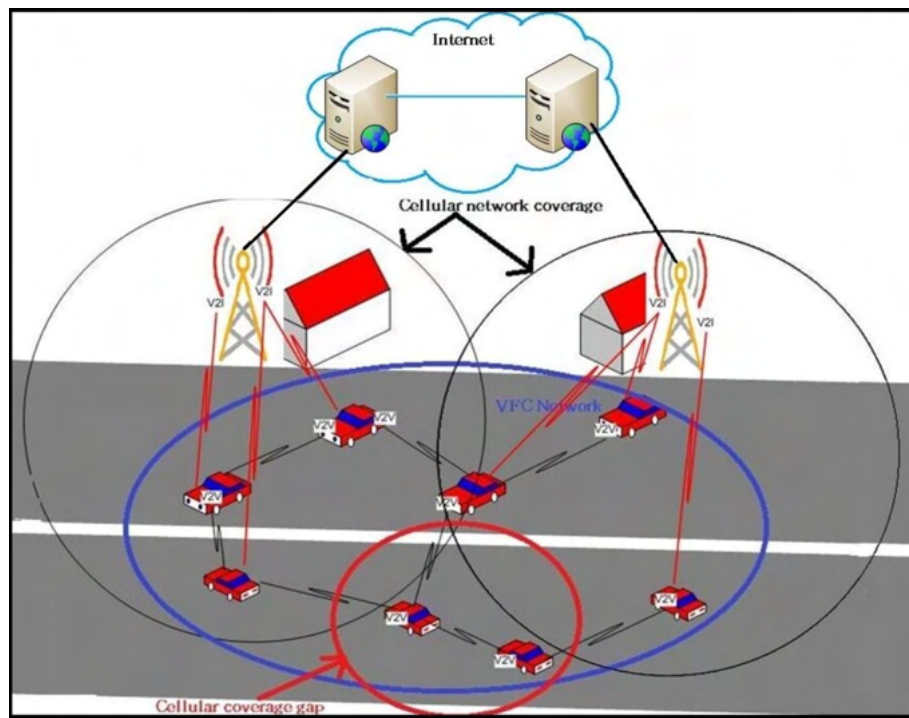


Figure 2.6 VFC network bridging the coverage gap of cellular network

driving, emergency message dissemination, etc. Latency-tolerant applications are entertainment, commercial advertising, route planning, etc.

2.8 Research Challenges

VFC is a promising area of research, particularly with ever-increasing onboard resources with modern vehicles. However, the following are major challenges

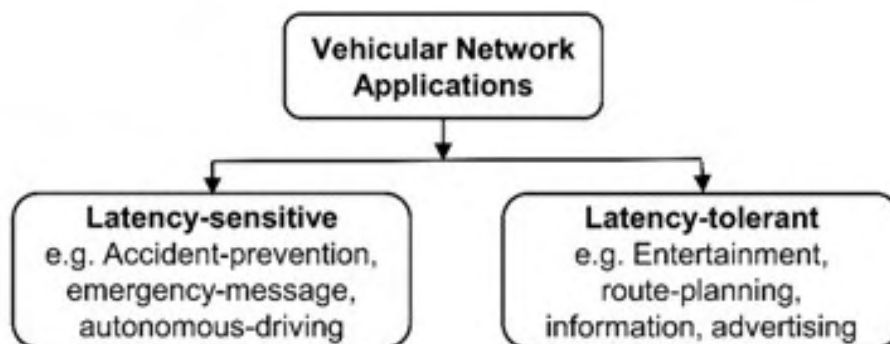


Figure 2.7 Classification of the vehicular networking applications

associated with VFC network implementation: -

Heterogeneity of nodes

Vehicular fog nodes in a VFC scenario are made of resources pooled from the available (i.e., moving, and parked) vehicular devices. These resources are heterogeneous with respect to storage, processing capability, and communication interfaces.

Mobility of nodes

Most of the VFC nodes are on the move. Thus, the availability as well as demand for resources at any given place, at any given point in time is highly dynamic (Meneguet et al., 2019; Mohanty, Kumar, Malakar, Vishwakarma, & Reza, 2018). VFC nodes communicate wirelessly, therefore, the availability and reliability of communication are also challenges. In addition, for such mobile nodes, a reliable handover mechanism between devices and applications must be provided.

Heterogeneous applications

Some VFC applications such as emergency message transmission services, although, require very small bandwidth, demand high reliability and ultra-low latency (Sathish & Kumar, 2020). Whereas, on the other hand, there are applications such as content sharing and AR/ VR applications, which may demand very high bandwidth, but may accept reasonable latency. Thus, there is a heterogeneity of applications that must be served by VFC nodes.

QoS support

Because of constrained processing capability, diversity of available processing resources, and application demand, implementing QoS in VFC is a challenging task.

2.8.1 Summary of Previously Presented Research Challenges

Vehicular fog computing is still at a nascent stage of development. It has increasingly drawn researchers in last 3 to 4 years. Various research challenges in vehicular fog computing have been discussed by the authors. Based on our literature survey, Table 2.5 summarizes the research challenges that have already been discussed by researchers.

Table 2.5 Summary of research challenges previously discussed by authors

Issues identified	Description	Insight
Mobility model	(Hou et al., 2016; Hoque & Hasan, 2019; Ge, Cheng, & Zhou, 2020)	Need proper protocols to handover between vehicular fog networks and other networks Vehicular networks are formed on the fly and have dynamic topology and membership. Hence, an efficient, and reliable handover mechanism is required.
Task Offloading	(S. Zhou et al., 2019; Liao, Zhou, Zhao, Ai, & Mumtaz, 2019)	Dynamic and effective task allocation tasks to the resources. A low-cost, improved, and asymmetric solution for supporting QoS.
Capacity analysis and resource management	(C. Zhu et al., 2018; P. Hu et al., 2017; Z. Zhou et al., 2020; X. Xiao et al., 2019)	Capacity analysis and management algorithms for real-time computing in the VFC network's capability. Manage resources in real-time, which is inevitable.
Price and motivation	(Z. Wang et al., 2018; Y. Wang, Xu, Zhou, Pervaiz, & Mumtaz, 2018; Peng, Ota, & Dong, 2020)	Suitable pricing and motivation methods for contributing devices. Nodes in VFC join the network voluntarily. Effective motivation and pricing mechanism must be in place.

Security, privacy and forensics	(P. Hu et al., 2017; J. Huang et al., 2020; Hoque & Hasan, 2019; Kong, Su, & Ma, 2020; Alamer & Basudan, 2020)	Security protocols and forensic analysis tools for participating vehicular nodes Privacy is everyone's concern, be it a service provider or service user. Rugged authentication, authorization, and forensic methods are desirable for establishing trust among users.
Communication protocols	(Y. Xiao & Zhu, 2017; Sodhro et al., 2020)	Strengthening of communication protocols to avoid radio channel interference Radio users are increasing exponentially, making interference a major concern. Strengthening radio protocols is necessary for efficiency.
Scaling	(Hoque & Hasan, 2019; Sayed & Kashkoush, 2020)	Up and down scaling of vehicular network based on the resources demand The demand for resources in the ITS domain is fluctuating. Hence system should scale the capacity dynamically.
Programming platform	(P. Hu et al., 2017)	Programming platform to support heterogeneous vehicular fog nodes This is a multidimensional area for programming. A programming platform to support its requirements needs to be developed/ customized.
Energy management	(P. Hu et al., 2017)	Energy management and optimization techniques for highly distributed architecture VFC is a highly distributed computing architecture. Energy management/optimization techniques are to be developed for such systems.

2.8.2 New Research Challenges Identified

In addition to the research challenges already discussed by various authors, the following important research challenges are identified from discussion and comprehensive analysis of research works.

Standardization of architecture

For practical implementation and application development for any upcoming technology, it is essential to have a well-established and standard architecture. Though many multi-tier architectures have been proposed for vehicular fog networks, it is yet to be standardized.

Optimal vehicular fog node selection

Vehicular fog nodes are a dynamic pool of collaborative, heterogeneous devices. Both, workload as well as resource availability in VFC vary dynamically, in time and space, because of the movement of nodes and end users. Hence, it is essential to select most suitable node dynamically, from available pool of nodes, to ensure QoS and optimize resource utilization.

Clustering techniques

Clustering techniques for VANET are center of attention of researchers. Suitable clustering techniques for vehicular fog networks need to integrate with other constraints like capability analysis, resource management, energy optimization, etc.

SLA and charging policies

A vehicular fog network is not an independent entity. It is an extension of and is designed to complement conventional fog networks. Service level agreements (SLAs) for implementing strict QoS achievement are to be developed. Well-defined billing and charging policies for resource sharing and pricing for fog providers and participating vehicular fog nodes (i.e., resource owners) to pool their assets are necessary.

2.9 Summary

ITS is the next big revolution globally and has tremendous scope to expand the role of vehicles, beyond mere transportation. VFC has a promising role in the ITS domain by opening avenues for much more efficient applications and services employing vehicles themselves as fog servers. Chapter begins by discussion on the issues on VFC implementation. We establish important characteristics of vehicular fog computing while discussing its architecture, and bring out its detailed comparison with conventional fog computing. We illustrate the inter-relationship between various characteristics of cloud, fog, and VFC, thus, unfolding their benefits and limitations concerning ITS. We have categorized the VFC applications and presented some novel applications and research challenges in this field. The capability of novel applications in VFC has been highlighted by explaining one example application use case in detail. This research work forms one of the most important survey in VFC, its applications in ITS domain. This survey provides new avenues for furthering research in this domain, by highlighting new applications and various unexplored research challenges.

Chapter 3

Multi Objective Modelling of Vehicular Fog Computing

3.1 Introduction

As part of this doctoral research work the task of resolving the identified research problem in Chapter 1 has been divided into two major parts. The first part consists of formulating the MOO mathematical model based on the literature survey, setting up the simulation environment and based on the simulation setup verifying the suitability of the established model. The vehicular fog computing architecture is modeled using a mathematical model. The modeling involves mathematical formulation of the individual components of the VFC model namely, latency, energy consumption, networking, communication, and propagation aspects. Based on the mathematical representation of the individual components the VFC is mapped into a multi-objective optimization (MOO) model to optimize latency. Developed MOO model is solved using a well-known optimization tool to validate the model, results are compared to latest research in the field. Above mentioned objectives have been presented in the Chapter 3 of the thesis. The second part of this doctoral research task consists of designing and developing suitable algorithm(s) for meeting the defined research objectives based on the established and verified MOO model. The algorithm(s) developed have been analyzed for the effect of various decision variables on the latency-sensitive applications in urban vehicular scenario. The parameters under study in this research work are latency, energy consumption and network bandwidth

consumption. The second part of the research work has been presented in the Chapter 4 of the thesis.

The novel contributions achieved in this chapter are the identification of the optimal VFN characteristics, establishing a MOO model encompassing the publisher-subscriber communication in 3-tier VFC, and solving the presented MOO model using the 'fmincon' solver. The organization of the chapter is as follows. Section 3.2 discusses system model architecture, and Section 3.3 discusses and establishes Optimal VFN characteristic. Sections 3.4 to 3.8 establish various mathematical models used in this research work. Section 3.9 presents the MOO model used to resolve the research problem presented in this research work. Section 3.10 describes the simulation environment and Section 3.11 presents the simulation results and their analysis. The Section 3.12 summarizes the Chapter.

3.2 System Model Architecture

Due to the emerging demand for new applications based on the requirements, the resource demand is also increasing and there is an urgent need to optimize the limited resources available in highly decentralized (Birhanie et al., 2020) vehicular networks. The vehicular fog network is formed on the go by vehicles joining and leaving dynamically. Not the same VFN remains in a place for longer because of movement of nodes. The demand for resources in vehicular networks is also very diverse and different applications demand different levels of QoS. The topology, resource demand, and resource availability at any place at a given time are unpredictable. In such a situation the assignment of tasks to available computing servers, while meeting QoS requirements of applications is a challenging problem(R. Hussain & Zeadally, 2019; J. Zhang et al., 2011; Hoque & Hasan, 2019). In (M. Hussain et al., 2020), a vehicular fog network planning scheme has been proposed, that decides the optimum node, and its capacity to minimize latency and energy consumed. In (M. Hussain et al., 2020), authors used a modified differential algorithm to minimize delay, and energy in VFN, and have depicted the advantages of the VFC system compared to cloud.

The generic, three-layered VFC architecture as proposed by the open fog consortium (Byers, 2017), used in this research work is shown in Fig 3.1. CDC provides a shareable, on-demand pool of resources. But it cannot support the latency demanded by delay-sensitive, real-time ITS applications (Haj Qasem, Abu-Srhan, Natourea, & Alzaghou, 2020), because of their location being far away from end-users. In VFC architecture fog network forms an intermediate layer, that are near to user. Computing resources in this layer are FCN located at routers, RSUs, switches, and gateways.

FCNs can support the latency requirement of latency-sensitive, real-time ITS applications. However, the deployment of FCNs to accommodate the dynamic workload of the vehicular network is very costly (Y. Xiao & Zhu, 2017) and it will result in sub-optimal utilization of installed fog resources. Thus, it either proves costly for service providers or, is not able to meet the end-user demand.

The bottom layer is the VFC layer, which comprises end-user devices like vehicle OBU, sensors, and passenger devices. Some of these vehicular OBUs having requisite computing resources (i.e., CPU and memory) and communication (i.e., link bandwidth) resources are selected as VFNs, which provide a pool of on-demand computational resources (X. Xu et al., 2020; Tang et al., 2020) for processing requests from nearby vehicular nodes. To simplify the discussion in this research work, the term vehicular node has been used to represent all devices in the VFC layer and it is assumed that there is seamless connectivity within and between all three, using V2X within the VFC and from VFC to fog layer and, using core IP network between fog and cloud layer.

In an ITS, the VFC architecture plays a very important role. It is a three-tier architecture. Tier III is a vehicular network or vehicular fog layer, where user devices and sensors reside. In this layer, tasks are generated, as well as selected OBUs (i.e., Vehicular Fog Nodes) accept tasks for processing, based on the priority of the task and resource availability. The vehicular fog layer has dynamic resource pooling capability. In case VFNs are not able to process the task because of the scarcity of resources, it is handed over to a higher layer, tier II (i.e., fog layer) for processing. The fog layer has dedicated fog servers with higher but fixed capacity installed at RSUs, base stations, gateways, and

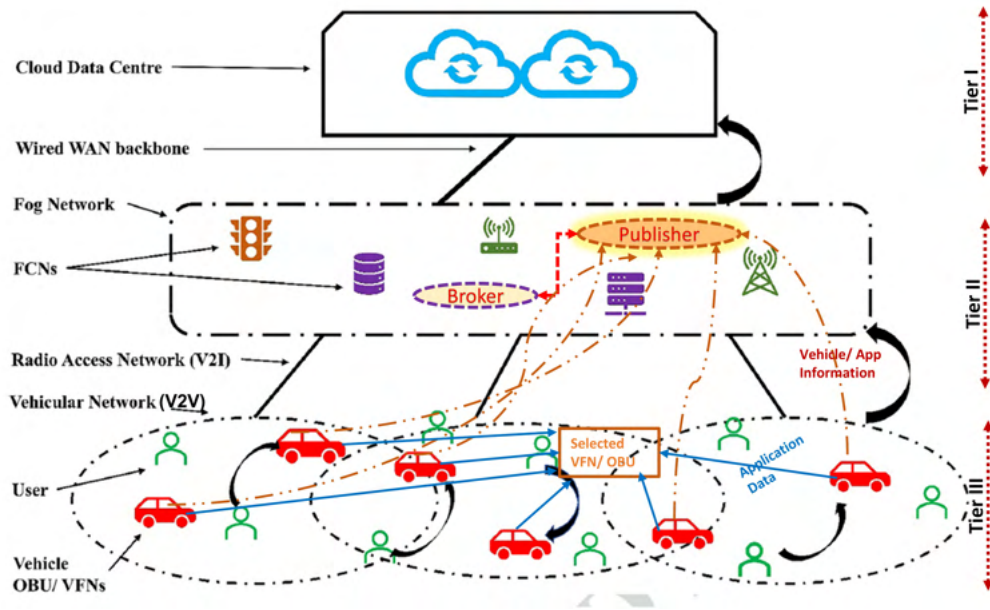


Figure 3.1 The architecture of VFC network: task processing and despatching to higher layers

routers to handle latency-sensitive tasks. The fog layer will process all remaining latency-sensitive tasks. All latency-tolerant tasks such as mass data processing and analysis, long term storage of massive amounts of data are presented to CDC for processing. As one moves from tier III to tier I, the processing capacity of nodes as well as latency increases (Binwal & Kapoor, 2022).

In an ITS, the VFC architecture plays a very important role. It is a three-tier architecture. Tier III is a vehicular network or vehicular fog layer, where user devices and sensors reside. In this layer, tasks are generated, as well as selected OBUs (i.e., Vehicular Fog Nodes) accept tasks for processing, as per priority of task and resources available. Vehicular fog layer has dynamic resource pooling capability. In case VFNs can not handle the task because of scarcity of resources, task is handed over to a higher layer, tier II. Middle layer has dedicated fog servers with higher but fixed capacity installed at RSUs, base stations, gateways, and routers to handle latency-sensitive tasks. The fog layer will process all remaining latency-sensitive tasks. All latency-tolerant tasks such as mass data processing and analysis, long term storage of massive amounts of data are presented to CDC for processing. As one moves from tier III to tier I, the processing capacity of nodes as well as latency increases (Binwal & Kapoor, 2022).

3.3 Optimal Vehicular Fog Node Characteristics

Vehicular network study has become a ubiquitous choice of academia and industry. The drawback of cloud that it is not able to meet the latency limits of latency-sensitive applications (Bonomi et al., 2012). Whereas fog and mobile fog networks are burdened by the burgeoning demand for new applications from vehicular users. Several of these applications are latency-sensitive. With the emerging demand for safety, entertainment, advertising, social networks, and traffic planning, applications among vehicular users, the demand for communication bandwidth and processing power near user nodes is increasing rapidly. The bandwidth and processing power available in the generic fog-cloud system is limited. Hence the concept of using moving vehicles themselves as fog nodes (i.e. VFN) to augment the capacity of existing fog-cloud infrastructure was proposed (Hou et al., 2016).

Vehicular fog networks have evolved from VANETs and thus share all the characteristics and benefits of VANETs. Important characteristics of VFC to differentiate it from fog and cloud are computation power, geographically distributed nodes, local decision-making, location awareness, low deployment cost, and low latency (Hou et al., 2016; Binwal & Kapoor, 2022). To enhance the QoS and safety on the road, it is imperative to minimize latency in VFC. The latency for a given request consists of three components, processing-latency, transmission-latency, and propagation-latency. Propagation latency varies with the distance between source and destination nodes. To minimize propagation latency component, a maximum number of the requests generated by the vehicular nodes must be processed within vehicular fog nodes, to minimize the distance from a data source to the processing node. To satisfy this condition, VFNs must have adequately high computational resources for the processing of requests. Availability of high processing capacity VFNs in turn reduces the computational latency, along with propagation latency. Similarly, for minimizing the transmission latency the transmission bandwidth between the data source and processing nodes must be high. Based on the above discussion, two important characteristics of the VFN are as follows:

First, the node should be capable of providing maximum processing capability (processor and memory), so that a minimum number of nodes can serve the network efficiently. This will alleviate the need to manage the network and reduce the delay caused by multiple-hop transmissions. It can be measured in terms of the excess processing capacity of the node or the normalized processor load factor, which can be represented as given below in equations (3.1) and (3.2).

$$Q_{v,i}^{cap} = \max Q_{w,i}^{cap}; \forall w \in W \quad (3.1)$$

Where, $Q_{v,i}^{cap}$ is the processing capacity of VFN and $Q_{w,i}^{cap}$ is the processing capacity of the vehicular node.

$$0 \leq \theta \leq 1 \quad (3.2)$$

Where, θ is the normalized processor load factor.

Second, the node should be equipped with multiple communication interfaces of high capacity for use in V2X communication. V2X communication is always a wireless communication system. Minimum delay and packet loss ratio are key measures of performance of V2X communication (Ejaz, Kumar, Ylianttila, & Harjula, 2020). The 5G-enabled VFC has been actively researched by many researchers for the past couple of years. The most important research problems of 5G technology in C-V2X are the modification required in the layer-2(MAC) and layer-1(physical) layers to make it suitable for the VFC (Alvi et al., 2022) and, security and privacy protection in VFC (Boualouache et al., 2021; X. Cheng et al., 2017). Therefore, currently, DSRC and 4G/LTE are the most common techniques used in V2X communication (S. Liu et al., 2018; Z. Xu, Li, Zhao, Zhang, & Wang, 2017).

3.4 Communication and Propagation Model

In this research work, two types of requests latency-sensitive and latency-tolerant have been considered. The DSRC uses point-to-point communication links for efficient V2V communication and, 4G/LTE is a cellular technique for long-

range V2I communication. The DSRC is proven to provide security, and privacy protection and, is more efficient for latency-sensitive safety-critical applications, and 4G/ LTE performs better for latency-tolerant non-real-time applications (Ejaz et al., 2020; Z. Xu et al., 2017). A judicious combination of both DSRC (i.e., short-range, but offers better QoS) and 4G/ LTE (i.e., long-range, but offers reduced QoS) is desirable for optimum VFN performance. Because of the above reasons, to prove the concept presented in this research work the combination of DSRC and 4G is used in V2X communication, instead of using still-developing technology 5G for C-V2X communication.

Publisher-subscriber communication model

This research work uses three-layer system design made up of the cloud, static fog, and vehicular fog layers as depicted in Fig 3.1. To simplify discussion, in this work (i.e., latency and resource optimization) by limiting the scope of the research work following assumptions are made. • The Rayleigh multi-path radio propagation model (Alouini & Goldsmith, 1997) is used for wireless communication scenarios (i.e., V2V and V2I). It assumes large-scale fading because of the multi-path effect in the presence of buildings, trees, etc. The fading distribution indicates the change in received signal strength of signal because of the multi-path effect. The path loss exponent value in the distribution indicates how far the radio signal reaches because of the multi-path fading. Typical values of path loss exponent are in the range of 2-5. This radio model has been used because it provides the most accurate traffic data generation using SUMO and OSM tools (Behrisch, Bieker, Simulation, & undefined 2011, n.d.) used in this research work for simulation. The publisher-subscriber model has been used as overlay network of underlying VFC communication. Discussion about underlying communication protocols and the issue of frequent disconnections/packet drops in wireless communication (i.e., V2V and V2I) is out of the scope of this research work.

Pub-sub model is an event-driven distributed message dissemination system that works asynchronously and does not require direct/dedicated connectivity. There are three basic building blocks of this model namely: publisher, subscriber, and message broker as depicted in Fig 3.1. The publisher is the message

provider. All the VNs register and publish the messages about their state /updates (i.e., location, speed, direction, communication interfaces, available processing capacity, etc.) to the publisher. The publisher publishes event-driven messages to different channels or topics. The subscribers are the VNs who wish to receive updates about specific topics. The role of a message broker is that of a middleman to enable communication and routing of messages from the publisher to subscribers. The pub-sub model depicted in Fig 3.1 ensures that any message about the VN state is immediately pushed from publisher to subscriber via a message broker.

The pub-sub is lightweight, loosely-coupled, and scalable model that provides robust and dynamic VFC services. In this research work, the well-known pub-sub model is implemented in a distributed fashion as depicted in Fig 3.1, for achieving computation offloading in the VFC scenario. The publisher and message broker elements have been implemented at FCN, which is static and seamlessly connected over a high bandwidth wired connection. The unique contribution of this work regarding pub-sub model implementation is that the scheduling, bandwidth management, and, allocation of computational resources to application requests, are performed at FCN, whereas, actual application request processing is performed at VFN. This scheme has been named computation offloading. It is a novel implementation of pub-sub paradigm in VFC for achieving computation offloading. This novel pub-sub implementation in the VFC scenario enhances QoS for latency-sensitive requests by, i) reducing the sag of network bandwidth, ii) preserving scarce computation resources at VFN for processing of application requests, and iii) reducing latency experienced by applications.

A latency-sensitive task request prodeced by vehicular node v for execution by a fog server (either a VFN or static fog node). In an urban scenario, all the vehicular nodes are connected to each other, and infrastructure nodes using DSRC and (or) LTE. A request is transmitted from the source (VN) to the destination processing node (VFN or FCN) using either V2V (for VFN) or V2I (for FCN) communication. The processing of a request from a vehicular node v to fog node f (i.e., pair v - f) is decided based on condition of minimization

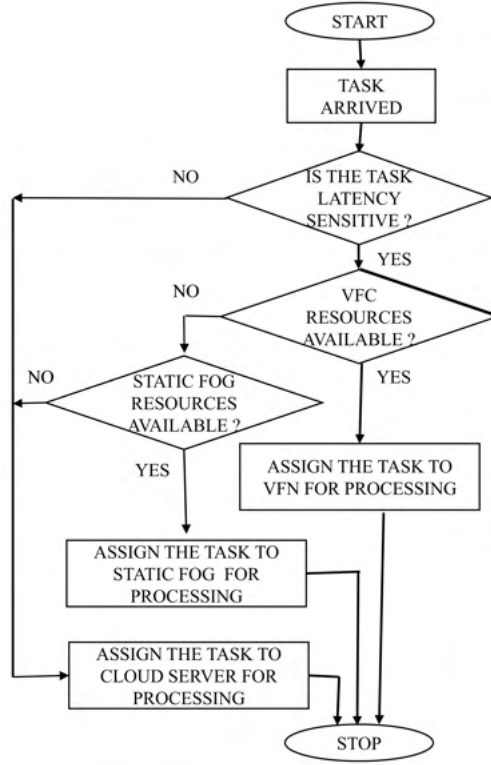


Figure 3.2 Flow chart of request processing

of overall energy consumption, subject to satisfaction of the constraints of QoS (i.e., maximum permissible latency δ_{max} for the request), processing capacity (Q) at the fog node, and link bandwidth between source and destination (b_v^f). Fig 3.2 illustrates the sequence of steps in a flowchart for processing a request in VFC.

3.5 Mobility Model

Pathway mobility model is used for simulation run of this research work (Bai & Krishnan, 2006). Pathway Model is effective method of fitting building and space constrictions into the geographic model, which also permits restricting the vehicular movement within pathways in the chosen map. We have used a real city map from Chandigarh, India (using OSM). The vertices are represented by the buildings, and the edges of the graph depict streets in between them. In contrast to the Random Way point model wherein devices move arbitrarily in any direction and length, the vehicular nodes in this model are permitted to travel only on the pathways. In this model, vehicular nodes are traversing in a

pseudo-random manner in predefined paths.

3.6 Networking Model

The vehicular fog network has been modeled as a three-tier architecture (Fig 6.1), as connected, undirected, weighted graph $G(V, E)$. Vertices in set $V(\cdot)$ comprise vehicular nodes (W), vehicular fog nodes (V), fog computing nodes (F), and cloud (C), i.e. $V(G) = W \cup V \cup F \cup C$. Edges $E(G)$ represent edges with weights $c = c_1, c_2, c_3, \dots$, where each c_l represents the cost (or delay) associated with the path between a pair of vertices in $V(G)$. Edges are made up of DSRC and 4G/LTE radio links. FCN to CDC communication is through Internet Protocol (IP) based core network (i.e., wired Wide Area Network links). Every vehicular node $w_i \in W$ generates workload (i.e., application requests) that needs to be processed by one of the computation units i.e., VFN, FCN, or CDC.

In this research work, the vehicular workload requests are categorized into two types, based on their latency requirement. Binary variable $L_b = 1$ indicates a latency-sensitive application and $L_b = 0$ indicates a latency-tolerant application. Workload requests generated in VFC could be simple data access requests or compute-intensive requests. Most of the latency-sensitive requests are compute-intensive (H. Li, Ota, & Dong, 2019), that need to be processed at VFN/fog nodes for meeting QoS. Examples of latency-sensitive applications in vehicular networks are safety applications such as turning speed indicators, collision distance prediction, accident information dissemination system, etc. and latency-tolerant applications are route planning information, entertainment applications, commercial advertising, etc. Latency-sensitive tasks must be processed either by VFN or the FCN, in this priority order subject to resource availability. Whereas latency-tolerant applications can be directly offloaded to CDC for processing, to optimally utilize the resources at VFN and FCN for latency-sensitive applications for ensuring the QoS.

Any application request will either be processed by the nearest VFN or will be despatched to an FCN, or further to a CDC for processing depending on value of binary variable L_b , and resources available at a given instance of time at VFN

and FCN. All those requests that could not be processed at VFN or FCN will be processed by CDC (assuming infinite resources at cloud server). Accordingly, latency model for different scenarios in the VFC network is developed in this research work.

3.7 Latency Model

The number of vehicular nodes in an area under consideration is random, and the number of workload requests generated by a vehicular node in a given time is very small and independent of other vehicular nodes. The workload arrival in the vehicular network follows Poisson's distribution. Poisson probability distribution represents the probability of multiple events happening in a predefined period provided, all events are independent and happen with known average rate (H. Hu, 2008). Each request has maximum permissible latency (δ_{max}) and the maximum permissible latency is pre-defined for type of application and(or) type of vehicle. Maximum permissible latency must be satisfied for meeting specified QoS e.g., maximum turning speed computation on a curve of the road is a safety-critical application that must satisfy the stringent latency limit δ_{max} . A latency-sensitive workload ($L_b = 1$) generated by a vehicular node will be sent to VFN, based on the instantaneous resource availability at the VFN. If it cannot be processed at VFN, it will be despatched to FCN for processing. In case of a latency-tolerant workload ($L_b = 0$), it will be directly dispatched to the cloud server. In either case, there is a transmission delay, propagation delay, and processing delay involved. Table 3.1 explains the symbols used in Equation (3.13) for representing total latency.

Each fog node processes one request at a time. A request is always presented to the node which is free to process a request. Hence, a request gets executed as soon as it arrives at VFN. For the FCN, there is more than one identical VM running in parallel, each can process one request at a time. QoS parameter δ_{max} is the maximum permissible latency defined for each task. The total latency for processing a request at the fog node consists of three elements:

Three different elements of latency are computing latency, transmission latency,

and propagation latency. Computing latency indicates total delay at the processing server (i.e., queuing time plus execution time). Latency experienced by a vehicular node to transmit data over channel is transmission delay. Propagation delay is delay in sending data from source to destination. A latency-sensitive request ($L_b = 1$) is processed at VFN. A latency-tolerant request ($L_b = 0$) is presented directly to CDC for processing. The VFN is modeled as M/M/1 queue with rate λ_i^a . The FCN is modeled as M/M/n queue arrival rate λ_f^a . For simplicity, it is assumed that FCN is made up of n homogeneous servers (or virtual machines), where n is a small number (i.e., $1 \leq n \leq 5$). The Erlang-C formula $C(n, \mu_f)$ is used in the FCN scenario. The Erlang-C formula is applicable for cases where because of the non-availability of resources the customer is placed in the queue till the time resources become available (S. Liu et al., 2018; Angus, 1946). The cloud is modeled as M/M/ ∞ queue.

3.7.1 Latency Element Definitions

The latency for processing of an application request is made up of various components, that are defined in the following paragraphs.

Computation latency

The latency in the processing of a given request at a fog node is termed computation latency.

Case : VFN

$$t_v^{comp} = \sum_{i=1}^w \frac{1}{L_b(Q_{v,i}^{cap} \cdot \theta - \rho_v \cdot \lambda_i^a)} \quad (3.3)$$

Where, θ is normalized processor workload, and $Q_{v,i}^{cap}$ is processing capability of v_i

Case : FCN

$$t_f^{comp} = \frac{\binom{n \cdot \mu_f}{n \cdot \lambda_f^s - \lambda_f^a}}{\binom{n \cdot \mu_f}{n \cdot \lambda_f^s - \lambda_f^a}} + \frac{1}{\lambda_f^s}, \quad \mu_f = \frac{\lambda_f^a}{n \cdot \lambda_f^s}, \quad (3.4)$$

and

$$(n \cdot \mu_f) = \frac{\binom{n \cdot \mu_f}{n!} \left(\frac{1}{1 - \mu_f} \right)}{\sum_{q=0}^{n-1} \frac{\binom{n \cdot \mu_f}{q!}}{q!} + \binom{n \cdot \mu_f}{n!} \left(\frac{1}{1 - \mu_f} \right)} \quad (3.5)$$

Where, λ_f^s is service rate at FCN

Case : CDC

$$t_c^{comp} = \frac{1}{\lambda_c^s} \quad (3.6)$$

Where, λ_c^s is service rate at CDC

Transmission latency

The latency in transmitting a given application request from the source (i.e., vehicular node) to the destination processing node (i.e., fog node) is transmission latency.

Case : VFN

$$t_{wv}^{tran} = \frac{\lambda_{w,v}^a \cdot \alpha}{B_{wv}}, \quad \text{and } \lambda_{w,v}^a = \rho_v \cdot \lambda_{i,v}^a \quad (3.7)$$

Where, α is workload size, and B_{wv} is the bandwidth of the W to V link

Case : FCN

$$t_{vf}^{tran} = \frac{\lambda_{w,f}^a \cdot \alpha}{B_{vf}}, \lambda_{w,f}^a = \rho_f \cdot \lambda_{i,v}^a, \quad and \quad \rho_f = 1 - \rho_v \quad (3.8)$$

Where, B_{vf} is the bandwidth of the V to F link

Case : CDC

$$t_{fc}^{tran} = \frac{\lambda_{w,c}^a \cdot \alpha}{B_{fc}}, \lambda_{w,c}^a = \rho_c \cdot \lambda_{i,v}^a, \quad and \quad \rho_c = 1 - \rho_f \quad (3.9)$$

Where, and B_{fc} is the bandwidth of the F to C link

Propagation latency

The latency experienced in reaching the request from a source VN at the location A (lat, long) to destination fog node B (lat, long) is propagation latency and is given by:

Case : VFN

$$t_{wv}^{prop} = \frac{D(w,v)}{c} \quad (3.10)$$

Where, $c = 3 \times 10^8 m/s$, and $D(w, v)$ is the Euclidean distance between W and V

Case : FCN

$$t_{vf}^{prop} = \frac{D(w,f)}{c} \quad (3.11)$$

Where, $D(w,f)$ is the Euclidean distance between W and F

Case : CDC

$$t_{fc}^{prop} = \frac{D(w,C)}{c} \quad (3.12)$$

$D(w,C)$ is Euclidean distance between w and C

Table 3.1 Different symbols used in equation (3.13) and their description

Symbol	Description
t_{wv}^{tran}	Data transmission time (for workload) between the vehicular node and VFN
t_{vf}^{tran}	Data transmission time (for workload) between VFN and FCN
t_{fc}^{tran}	Data transmission time (for workload) between FCN and CDC
t_{wv}^{prop}	Propagation time from vehicular node to VFN
t_{vf}^{prop}	Propagation time from VFN to FCN
t_{fc}^{prop}	Propagation time from FCN to CDC
t_v^{comp}	Computation time at VFN
t_f^{comp}	Computation time at FCN
t_c^{comp}	Computation time at CDC

The total latency encountered by an application request is given by Equation (3.13).

$$\delta_s = \rho_v(t_{wv}^{tran} + t_{wv}^{prop} + t_v^{comp}) + (1 - \rho_v) \{ L_b(t_{vf}^{tran} + t_{vf}^{prop} + t_f^{comp}) + (1 - L_b)(t_{fc}^{tran} + t_{fc}^{prop} + t_c^{comp}) \} \quad (3.13)$$

Where ρ_v is the probability of application being processed by VFN. Whenever a request is received for processing, an estimated latency (δ_{est}) for processing it at VFN is computed based on the request parameters and instantaneous processor load (θ). The probability of processing at VFN is given by equation (3.14) below:

$$\rho_v = Prob (\delta_{est} \leq \delta_{max}) \quad \forall w \in W \quad (3.14)$$

Where, δ_{est} and δ_{max} are the estimated latency for the set of selected pairs of source-processing nodes, and the maximum permissible latency for the application request, respectively. The value of δ_{est} is chosen empirically (random values between 25 and 75). Calculation of δ_{est} is provided by the ACT-DW algorithm proposed as part of this research work (discussed in Chapter 4). Thus, $(1 - \rho_v)$ gives the probability of the request being despatched to FCN or CDC.

Requests generated from vehicular nodes follow a Poisson distribution with a rate λ_i^a . Net traffic request arrival rate at VFN at any given instance is given by equation (3.15):

$$\lambda_{i,v}^a = \sum_{i=1}^w \lambda_i^a \quad \forall w \in W \quad (3.15)$$

3.8 Energy Model

Total energy consumption for the processing of a request at the fog node (VFN or FCN) consists of the following two elements.

Transmission energy

Switching-ON time (T_{on}) the time when node is transmitting or forwarding data. Off-time (T_{off}) is the time when node is listening. Accordingly, duty cycle (D) of is defined:

$$D = T_{on} / (T_{on} + T_{off}) \quad (3.16)$$

Energy consumed in the sending data packet from node-v to node-f consists of three elements namely trans energy (E_{trx}), forwarding energy (E_{fwd}), and listening energy (E_{lst}) as given below (Ghosh, Das, & Neogy, 2016).

$$E_{trx} = \alpha * 117 * 10^{-9} + \alpha * 1.7 * 10^{-6} + (D * \alpha * \epsilon * d^n) \quad (3.17)$$

$$E_{fwd} = D(\alpha * E + \alpha * \epsilon * d^n) \quad (3.18)$$

$$E_{lst} = (1 - D) * 570 * 10^{-6} \quad (3.19)$$

Where, α is size of data, d is inter nodal distance, n is pathloss-exponent ($n=3$), and ϵ is permittivity, ($\epsilon = 8 : 854 pJ/bit/m^2$) (Ghosh, Neogy, Das, & Mehta, 2017).

Total transmission energy is sum of all the three elements given above. Expression for transmission energy consumption E_t^v at a vehicular node is:

$$E_t^v = E_{trx} + E_{fwd} + E_{lst} \quad (3.20)$$

The transmission energy model uses a state-based energy model. Power consumption of radio node is decided by state of the radio device. Different states and their power consumption are depicted in Table 3.2.

Table 3.2 States of node and corresponding power consumption

State of node	Power consumption (mW)
Off	0
Sleep	1
Idle	2
Receiver busy	5
Receiver receiving	10
Transmitter transmitting	100

Computational energy

Energy consumed during computation process of a request of size α at the fog node is given by:

$$E_c^v = (C_f \cdot \alpha) / \psi_f \quad (3.21)$$

Where, ψ_f is the processing rate, which indicated the size of data processed per unit of time, C_f indicates the power rating of the processor.

Thus, total energy consumed in processing a latency-sensitive request of size α generated by vehicular node v at VFN f is:

$$E = (E_t^v + E_c^v) \quad (3.22)$$

3.9 Multi Objective Optimization (MOO) Model

Because of movement of vehicles, workload in VFC is generated dynamically. To meet latency expected for real-time, latency-sensitive requests generic fog-cloud system is inefficient, due to limited, and fixed hardware assets availability in fog and the cloud system. Thus, a conventional fog-cloud system is not suitable for latency-sensitive request processing. Whereas, vehicular fog networks permit a promising resource for such latency-sensitive, real-time, and dynamic workload, because of the on-demand, collaborative nature of resource pooling capability in VFC. Thus, there is a need for the VFC system, that offers an optimal solution for minimizing latency. This depends on two major aspects namely, the network connectivity of nodes, and latency-reducing strategies (Tiwari & Kumar, 2015). Therefore, in this research work, the VFC system has been modeled as a combination of two different sub-components, the networking model and the latency model. The networking model describes the connectivity of different nodes along with associated links, representing network topology. The latency model represents different factors of latency encountered by workload, for processing under different processing scenarios.

3.9.1 Assumptions and Pre-requisites

In this research work, the following pre-requisites are taken while formulating the VFC model.

- Firstly, there is seamless connectivity amongst all three layers presented in Fig. 3.1 using V2X and IP connectivity.
- Secondly, the arrival rate of vehicles at any point in time follows Normal distribution, and that of workload in vehicular fog nodes follows Poisson's distribution.
- Thirdly, the workload generated is atomic concerning task execution i.e., any given workload assigned to one VFN is executed completely at that VFN, or the entire workload is rejected by the node, and an alternative computing node is assigned.
- Lastly, each real-time request has a permissible latency limit that is pre-defined.

3.9.2 QoS Satisfaction Ratio

QoS satisfaction ratio for the latency-sensitive request is ratio of the number of applications for whom overall latency is smaller or equal to maximum-permissible-latency, to total number of latency-sensitive requests. QoS satisfaction ratio is expressed by Equations (3.23) and (3.24).

$$\gamma = \sum_{i=1}^n \gamma_i \quad \forall r \in R \quad (3.23)$$

$$\text{Where } \gamma_i = \begin{cases} 1 & \text{for } \delta_s \leq \delta_{max} \\ 0 & \text{for } \delta_s > \delta_{max} \end{cases} \quad (3.24)$$

3.9.3 Multi Objective Optimization Model Definition

In this part of the research work, the problem of minimization of latency in VFN is formulated as a MOO model. The MOO presented in this paper is aimed at solving the problem of assigning a dynamically generated workload

to a dynamically identified computing node (i.e., VFN). The model consists of three important elements namely objective function, decision variables, and constraints. The problem of minimizing the latency, while assigning the workload generated by a vehicular node to one of the processing node(s) in the VFC scenario can be expressed in the form of an optimization model expressed by equations (3.25) to (3.31) as follows:

$$\begin{aligned} & \text{Minimize} \\ & \forall w, \forall v, \forall f \quad (\delta_s) \end{aligned} \quad (3.25)$$

Subject to constraints:

$$\lambda_c^s > \lambda_f^s \quad (3.26)$$

$$\lambda_f^s \geq \lambda_v^s \quad (3.27)$$

$$\lambda_{i,v}^a \geq \lambda_v^s \quad (3.28)$$

$$Q_{v,i}^{cap} = \max Q_{w,i}^{cap} ; \forall w \in W \quad (3.29)$$

$$\sum_{w \in W} Q_w^{cap} \leq \sum_{v \in V} Q_v^{cap} \quad (3.30)$$

$$\sum_{v \in V} Q_v^{cap} \leq \sum_{f \in F} Q_f^{cap} \quad (3.31)$$

The optimization equations represent the multi-variable non-linear constrained MOO model. Objective function to be optimized is given in (3.25). Constraints in (3.26) and (3.27) are in-equality constraints that define upper and lower bounds on service rates at CDC, FCN, and VFN. Constraint (3.28) puts a limit on the maximum arrival rate at VFN so that VFN capacity is not swayed by excess demand. Decision variables are $\rho(v)$, θ and, α as defined in equations (3.3) to (3.14). To optimally utilize available resources for meeting maximum la-

tency constraints (i.e., QoS) defined by Equations (3.23) and (3.24), the problem of assigning the workload request to dynamically available computing resources, as formulated in (3.25) to (3.31) needs to be solved.

3.10 Simulation Environment

The MOO model presented above represents a complex problem involving many parameters. It is not feasible to solve the proposed MOO model using the direct method within practical time to be suitable for latency-sensitive applications. The research work proposes a MOO problem and its solution by an Interior-point optimization algorithm, with a sparse linear constraint matrix. For this purpose, a well-known optimization tool ‘fmincon’ from MATLAB optimization toolbox (Coleman, Branch, & Grace, 2009) is used. It is used for the minimization of multivariable functions starting at user-defined initial values between defined lower-bounds and upper-bounds. Constraint input can be linear and non-linear equality and (or) in-equalities. It permits the selection of various options for optimization runs.

The most important advantage of the ‘fmincon’ solver is that all constraints and input variables can be defined in matrix form. Thus, the accuracy of the inputs is ensured, and the optimization results are achieved faster. In the present research work, this tool solves the proposed MOO model for a local optimum solution. The solution proposed follows the classical divide-and-conquer strategy. Here the latency minimization is achieved by limiting task assignment to the area of radio coverage of the vehicular nodes (i.e., in single-hop communication), beyond which the workload is assigned to the FCNs or the Cloud. The local optimum solution is obtained for each of the small clusters (i.e., coverage area). In this case, the global optimum solution is not desirable, keeping in mind latency-tolerant task execution in close vicinity and infinite computing capacity in the cloud for latency-tolerant requests.

Table 3.3 gives various input arguments used by the solver for solving the simulation. These parameters have been chosen to get optimum results for a diverse variety of simulations. For the problem at hand, there is no known/

reliable source of data available for setting up simulations. Table 3.4 represents the values of constraints (i.e., lower and upper bounds) used in the multi-variable constrained optimization model. The bandwidth and distance parameters chosen are based on the real-world specifications of LTE and DSRC radio (S. Liu et al., 2018; Z. Xu et al., 2017; Bai & Krishnan, 2006). The number of vehicles and workload size are chosen empirically. Based on the specifications of problem the linear constraint matrix for MOO is sparse. Thus, the Interior-point algorithm chosen, for solving the MOO model is the most suitable, which handles large, and sparse problems. This algorithm also satisfies lower and upper bounds at all iterations.

For simulation, the range for workload sizes is from 300 to 1000, the probability of processing at VFN is from 0.2 to 0.9, and the number of VFN is 4 to 6 for the vehicular fleet size from 20 to 100. In this research work, extensive simulations were conducted for different values of workload sizes, the number of vehicles, the probability of processing at VFN, and the rate of request arrival in VFC to evaluate latency and QoS satisfaction ratio. The simulations are run on Intel(R) Core (TM) i5-1035G1 laptop with 8 GB RAM.

Table 3.3 Different input arguments used with 'fmincon' solver

Parameter name	value
Optimization algorithm	Interior-point
Linear constraint matrix	Sparse matrix
Max number of iterations	5000
Max function evaluations	15000
Tolerance on constraint-violation	1e-06

Table 3.4 Variables and the values of constraints

Variable name	Lower bound	Upper bound
Vehicle-VFN bandwidth (kbps)	250	3000
VFN-FCN bandwidth (kbps)	500	5000
FCN-CDC bandwidth (kbps)	10000	100000
Distance of VFN (m)	0	300
Distance of FCN (m)	400	5000
Distance of CDC (m)	10000	∞
Number of vehicles	20	100
Workload size (k bytes)	300	1000
Probability of processing	0.2	0.9
Number of VFN	4	6

To ascertain the statistical validation of the simulation data (i.e. variation in the dependent variable), simulation data is offered a one-way ANOVA test (Keselman et al., 1998; Brooks & Johanson, 2011). A summary of one-way ANOVA results for $\sigma = 0.05$ are presented in Table 3.5. The ANOVA results exhibit $F \geq F_{crit}$ and $P - value \leq \sigma$, thus null hypothesis that the variance between data groups isn't due to random chance is rejected with 95 percent confidence. Hence, this proves the statistical validity of the simulation data.

Table 3.5 Summary of one-way ANOVA test on simulation data

Source of variation	SS	df	MS	F	P-value	F-crit
Between Groups	128.36253	14	9.16875	3.71369	0.00125	2.03742
Within Groups	74.067059	30	2.46890			
Total	202.42959	44				

3.11 Simulation Results and Analysis

To validate the proposed MOO model, an optimization run was carried out to compare the proposed system with the generic fog-cloud, and two state-of-the-art techniques $ECOS$ and ETE_{FN} . Fig 3.3 shows latency performance of our technique in comparison to generic fog-cloud, $ECOS$, and ETE_{FN} for different workload sizes from 300 to 1000 bytes. The results obtained indicate very high values of latency (from 72.9926 to 240.6921 seconds) for the generic fog-cloud system, 0.7082 to 15.9874 seconds for $ECOS$, 0.5806 to 10.9874 seconds for ETE_{FN} , as compared to 0.2132 to 0.7083 seconds for the proposed scheme. The sharp increase in the latency values with size of workload in $ECOS$ scheme is due to the fact that as the size of workload increases more number of the tasks are processed at cloud layer (because of the scarcity of resources at VFN). Whereas, in the proposed scheme latency is kept minimum by processing all the requests at VFN (by pooling excess resources on real-time basis). Similar is the case with ETE_{FN} scheme, wherein the tasks are offloaded to cloud because of the nodes going out of RSU range. More tasks are processed at cloud as the workload size, hence, the processing time increases. Thus much higher latency in ETE_{FN} compared to proposed scheme, wherein dynamic resource provisioning is implemented, ensuring all the latency-sensitive tasks being processed at VFN. There is on average 350 times higher latency using the generic fog-cloud scheme, as compared to the proposed scheme. Latency values obtained for this scheme are much lower in comparison to $ECOS$ and ETE_{FN} . There is a lin-

ear increase in latency with the increase in workload size for all the schemes as depicted in Fig 3.3, which is attributed to the transmission delay increasing linearly with the size of the workload.

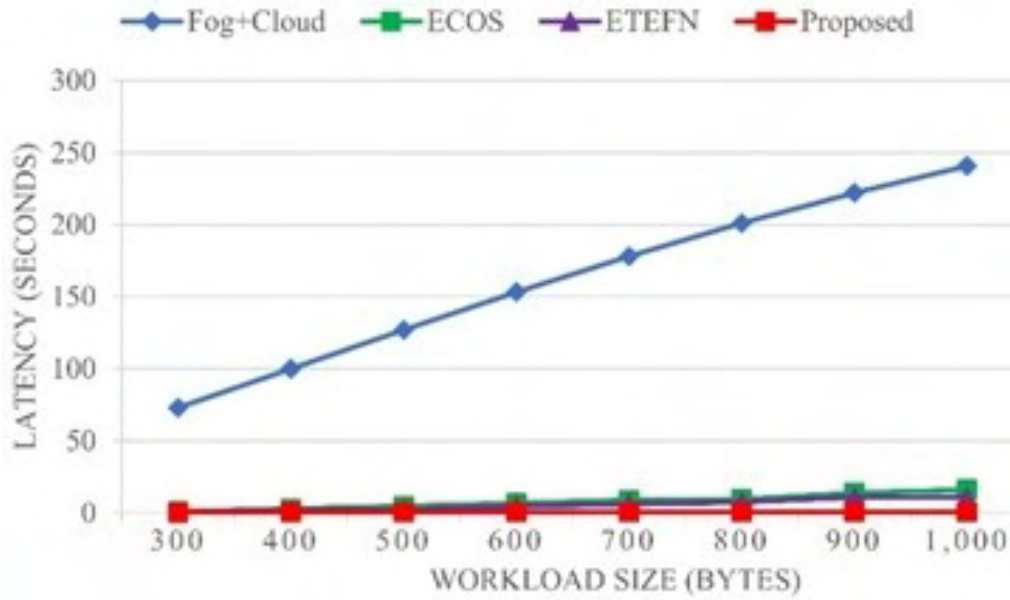


Figure 3.3 Plot of the latency variation with workload sizes

For many latency-sensitive requests in ITS e.g., safety applications, it is very important to meet the maximum latency constraint. If the application processing does not satisfy this constraint, the results of processing may not be of any use to the application/ user. The QoS satisfaction ratio for latency-sensitive requests is computed, as per Equations (3.23) and (3.24). Various parameters used for computing the QoS satisfaction ratio are given in Table 3.6. Typical values of maximum permissible latency in latency-sensitive applications in VFC range from 70 to 150 ms (Zadobrischi & Damian, 2021; Yadav et al., 2020)]. In this simulation maximum permissible latency value of 80 ms is used.

For many latency-sensitive requests in ITS e.g., safety applications, it is very important to meet the maximum latency constraint. If the application processing does not satisfy this constraint, the results of processing may not be of any use to the application/ user. The QoS satisfaction ratio for latency-sensitive requests is computed, as per Equations(3.23) and (3.24). Various parameters used for computing the QoS satisfaction ratio are given in Table 3.6. Typical values of maximum permissible latency in latency-sensitive applications in VFC range

from 70 to 150 ms (Zadobrischi & Damian, 2021; Yadav et al., 2020). In this simulation maximum permissible latency value of 80 ms is used.

Table 3.6 Parameter values used for calculating the QoS satisfaction ratio

Parameter Name	Value
No of requests	100
Request size (k Bytes)	300-1000
Request arrival rate	0.5 – 5.0
Number of VFNs	4-6
Probability of processing at VFN	0.7
Maximum latency (ms)	80

Fig 3.4 depicts a plot of the QoS satisfaction ratio for latency-sensitive requests, for the proposed VFC system, *ECOS*, *ETE_{FN}*, and generic fog-cloud system concerning arrival rate. The arrival rate at VFN follows Poisson distribution, is defined in Equation (3.15). The request arrival rate is a random variable and, is dependent on the number of vehicles W and λ_i^a . Arrival rate signifies workload on the VFC system. For this simulation, a set of 100 latency-sensitive requests with a request arrival rate of 0.5 to 5.0 were run for the proposed scheme, *ECOS*, *ETE_{FN}*, and generic fog-cloud system.

It is evident that with rise in arrival rate the QoS satisfaction ratio declines sharply (i.e., latency increases sharply) in the case of the generic fog-cloud system. However, the QoS satisfaction ratio for *ECOS* and *ETE_{FN}* schemes drops from 93 to 83 and 94 to 78 percent, respectively. Whereas, in the case of the proposed VFC system, the QoS satisfaction ratio remains approximately constant (i.e., from 96 to 93 percent ($\pm 3\%$)). The improved QoS satisfaction ratio of the proposed system is because, as the arrival rate and (or) number of vehicles increase, a greater number of computational nodes (i.e., VFN) are available for an effective increase in the computational capacity of the VFC. Hence, presented VFC system is adaptive to dynamic workload in providing an enhanced

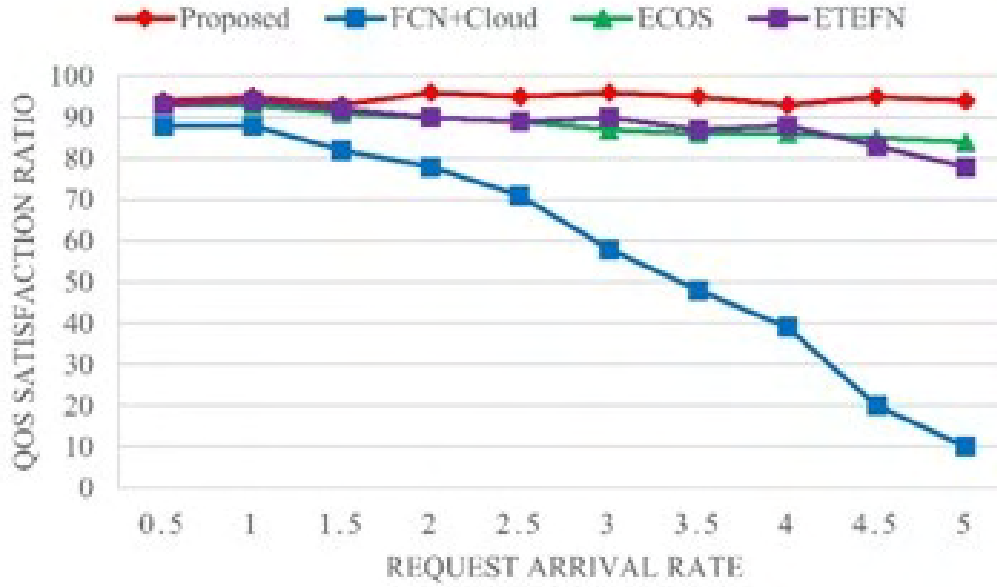


Figure 3.4 Variation of QoS satisfaction ratio with request arrival rate

QoS satisfaction ratio. Whereas, *ECOS* system uses offloading to cloud (to minimize energy consumption) with increased workload requests, resulting in higher latency or reduced QoS satisfaction ratio. Similarly, *ETEFN* schemes uses RSU coverage range and remaining processing time as the parameters for offloading to cloud. As the arrival rate increases, due to increased remaining processing time at VFN, more number of tasks are offloaded to cloud, increasing the latency or reducing the QoS satisfaction ratio.

To minimize latency, a request must be processed at VFN rather than FCN or cloud. The probability of processing a request at VFN as per Equations (3.14) is dependent on max permissible latency, the size of the request, and the instantaneous value of processor load factor at VFN. The proposed scheme variation in the latency with the probability of processing at VFN for three different vehicle fleet sizes of 20, 50, and 100 is depicted in Fig 3.5 (number of VFN = 4) and Fig 3.6 (number of VFN = 6), respectively. Results indicate, with the vehicular fleet size becoming large, there is an increase in total workload at VFN as per Equation (3.15). Minimum latency is obtained at 0.45, 0.65, and 0.80 (for the number of VFN = 4) and at 0.60, 0.65, and 0.70 (for the number of VFN = 6) values of probability of workload processing at VFN, for 100, 50, and 20 vehicle fleet sizes, respectively. With the increased workload due to the larger vehicle

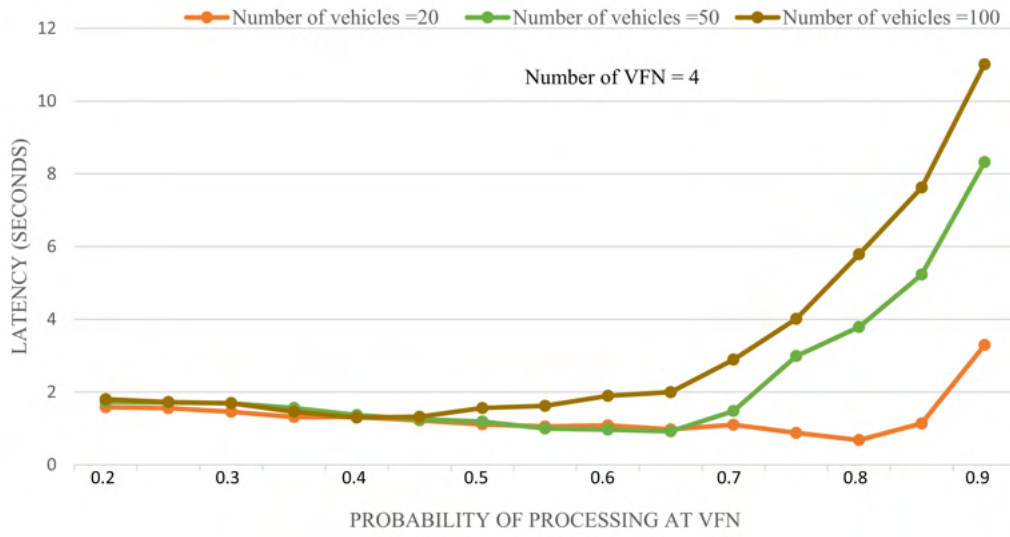


Figure 3.5 Variation in latency with the probability of processing a request at VFN for different fleet sizes (number of VFN = 4)

fleet size, the processor loading factor of VFN increases, and it eventually results in increased overall latency. For an increased number of vehicles, the latency increases exponentially, with early saturation of resources at VFN for a higher fleet size of 100, 50, and 20, respectively. Results prove that increasing vehicular fleet size, makes a proportionate increase in the offloading of requests to FCN is necessitated, despite the increase in the computational capacity of VFC (by an increase in the number of VFN from 4 to 6). The flattening (i.e., delayed rise) of the curve in Fig 3.6 as compared to Fig 3.5 proves the merit of the dynamic capacity increase of the proposed system. As the number of VFNs increases from 4 to 6, QoS satisfaction ratio is enhanced by approximately 15 percent, as measured by the increased flatness of the latency curve in Fig 3.6.

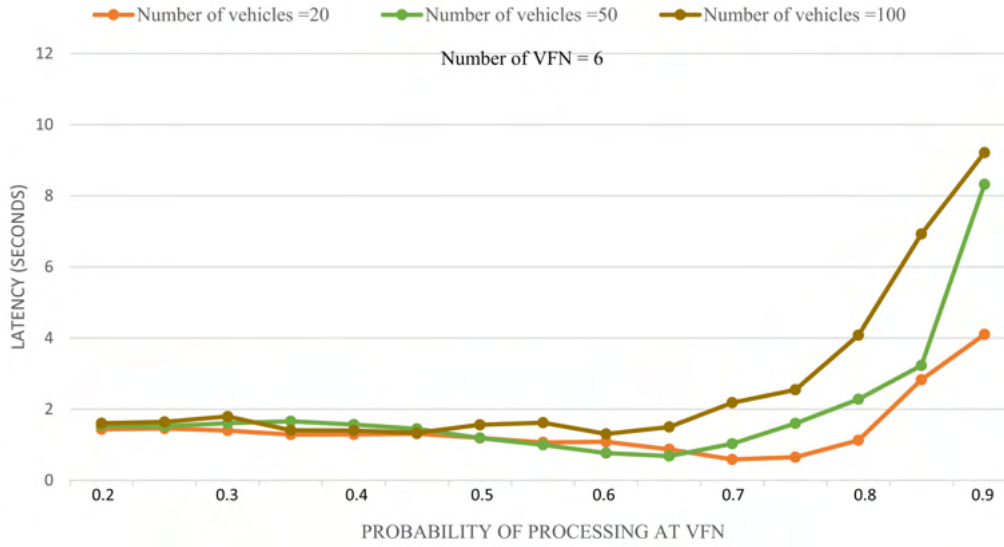


Figure 3.6 Variation in latency with the probability of processing a request at VFN for different fleet sizes (number of VFN = 6)

3.12 Summary

Minimizing latency and ensuring QoS compliance for latency-sensitive, real-time requests in ITS is the most important concern for ensuring the safety of vehicles and road users. Utilizing VFNs for the processing of latency-sensitive requests generated by vehicular nodes is a promising area for researchers aiming at achieving goal of minimizing the latency thereby ensuring QoS satisfaction in vehicular networks. In this chapter of the research work, the two most important characteristics of optimal VFN (i.e., high computational capability and, high bandwidth communication interfaces) have been identified based on a detailed research investigation. The mathematical modeling of VFC in the form of a MOO has been presented and solved using a well-known optimization tool. An extensive simulation was conducted using real-world values of decision variables to examine the optimum values of various decision variables for minimizing the overall latency and maximizing QoS satisfaction for latency-sensitive requests. Simulation proves superiority of the presented solution, as compared to fog-cloud, $ECOS$, and ETE_{FN} . Latency is much lower for presented solution in comparison to the generic fog cloud, $ECOS$, and ETE_{FN} . QoS satisfaction ratio of the proposed technique is maintained from 96 to 93 percent (within $\pm 3\%$), as compared to lower values and much wider variations in comparable schemes. The analysis of simulation results in this chapter proves suitability

of the presented technique for the selection of vehicular fog nodes in a VFC environment, where workload generation is dynamic and the conventional fog-cloud technique is not able to satisfy latency and QoS requirements, due to fixed computational capacity.

Chapter 4

A Lightweight Energy-efficient Technique for QoS Enhancement in Urban VFC for Intelligent Transportation System

4.1 Introduction

This chapter of the thesis presents the novel technique for enhancing the QoS established in this doctoral research. A heuristics based Lightweight Energy-efficient Algorithm for optimal fog node selection for processing of the latency-sensitive tasks has been presented. The major contributions of this chapter are as follows. Firstly, a heuristics based LEAQoS algorithm for selection of optimal VFN. Secondly, an adaptive capacity tuning algorithm (ACTDW) for real-time processing capacity adaptation of the VFC system. Finally, a concept of compute offloading using pub-sub model so as to minimize the resource consumption. Rest of the chapter is organized as follows. Section 4.2 covers the background and important motivation to undertake this doctoral research work. Section 4.3 presents system model architecture. Section 4.4 describes the proposed novel techniques overview, the LEAQoS and ACTDW algorithms. Section 4.5 explains important contributions of the algorithm(s), and Section 4.6 summarizes the chapter.

The VFC has a significant role in ITS implementation. In recent years researchers have introduced several new uses of VFC. Some of them are smart parking reservation system (Y. Zhang et al., 2019; Tiwari et al., 2019; Khan et al., 2018; Haj Qasem et al., 2020), crowd sourcing of road and traffic conditions (C. Zhu et al., 2018), smart real-time traffic management system (Z. Wang et al., 2018), and emergency message delivery system (Hou et al., 2016; B. Liu et al., 2021). The applications in connected vehicles are classified into latency-sensitive and latency-tolerant, based on value of maximum permissible latency limits. A latency-sensitive request requires processing on a real-time basis and generally has safety implications associated with it. If the results of processing /query to a latency-sensitive request are not available within well-defined time limits, they may not be useful, anymore. A large number of applications in ITS are latency-sensitive (Tang et al., 2020). A detailed list of existing and potential latency-sensitive applications in the connected vehicle domain is depicted in Table 4.1.

Cloud computing offers enormous computing power. It has essential drawback of offering high delays due to large commuting distances involved (Shrestha, Bajracharya, & Nam, 2018; Tiwari, Mittal, Garg, & Kumar, 2022). The idea of fog was presented (Bonomi et al., 2012) to decrease the delays in processing by fetching processing resources nearer to the end-user. In ITS fog micro servers are installed at RSU, base stations, routers, and switches (Consortium & Working, 2017). Static fog nodes (FCN) are infrastructure-based, have fixed capacity, and involve operating costs. Hence, FCN is not a cost-effective, and scalable solution for vehicular applications.

Table 4.1 List of latency-sensitive applications in the connected vehicle domain

Application	Description
Autonomous driving	Self-driving /driverless connected vehicles
Assisted driving	The ability for a car to sense, anticipate, and alert about potential dangers long before the driver senses it

Accident detection and reporting	Automatic accident detecting and reporting system
Accident prediction and analysis	Sensors on board vehicles and on-road contribute data continuously towards analysis. These results are used to predict and prevent accidents
Smart emergency response system	Automatic quick road emergency response system
Smart /adaptive traffic light	Adaptive traffic light system to intelligently route vehicular traffic
Leasing of excess resources to neighboring vehicles	Co-operative computing environment, wherein, underutilized resources of vehicles are leased to neighboring vehicles on-demand basis
Task offloading from fog/edge networks to vehicular fog network	Compute resources at the edge or fog network can be supplemented by offloading tasks to nearby VFC network
Task offloading from cellular network /Mobile Edge Computing (MEC) to VFC	Offloading of tasks from nearby infrastructure networks to VFC network utilizing V2I communication
Detecting and bridging mobile broadband coverage gaps	Augmenting internet connectivity for end-users / sensors, by detecting and bridging mobile broadband (i.e., Wi-Fi/ 3G/ 4G/ 5G) coverage holes

In (Datta, Da Costa, Harri, & Bonnet, 2016), the authors brought out that the integration of connected vehicles to the transportation system along with the IoT can unleash an efficient and sustainable ITS environment. The authors brought out the challenge of finding a replacement to the cloud for latency-sensitive applications. With the continuous increase in latency-sensitive applications, the requirement for an alternative computing platform suitable for such applications is inevitable.

Advent of the VFC bring about the challenges in its implementation e.g., QoS improvement, security and privacy, and efficient resource management have been highlighted by the authors, in (Menon & Prathap, 2017). The VFC, as a new computing model, poses problems that are to be solved for its implementation. In (C. Huang, Lu, & Choo, 2017), the authors brought out the capabilities of the VFC to address drawbacks of static fog. The authors also proposed the three-tier architecture of the VFC supported by application use-case for smart vehicles. The VFC model has the potential to solve the latency and other related issues of real-time applications faced by vehicular networks.

One vital feature of vehicular networks is dynamic workload generation. Because of certain unpredictable conditions on the road e.g., accidents, road conditions, weather conditions, certain events in the locality, traffic jams, etc., the traffic mobility pattern, and accordingly, the application demand by vehicular nodes may spurt suddenly. A prominent example of this is the morning and evening traffic conditions in urban areas during peak traffic hours. TThe restriction of FCN is, due to unpredictable road conditions or peak traffic hours, the processing power needed may not be available, because of surge in demand. To alleviate this situation, if we choose to deploy extra servers (taking into account full load), it results in underutilization of expensive resources most of the time (Hou et al., 2016; M. Hussain et al., 2020). In VFC, need to install extra hardware is countered by employing the excess hardware capability of the vehicle's OBU, on-demand (Tiwari & Kumar, 2015; X. Xiao et al., 2019; Tiwari & Kumar, 2016). Therefore, there is no added cost incurred in employing VFN.

4.2 Motivation to Undertake Research Work

Fluctuating demand and supply of processing power are two important characteristics of vehicular networks (Hou et al., 2016; Z. Zhou et al., 2020). Issue of minimizing delay, QoS improvement, and energy consumption in VFC has been researched actively for the last few years (K. Zhang et al., 2020; Alharbi et al., 2020; Mekki et al., 2020; Mahmud et al., 2020; Zadobrischi & Damian,

2021). With the smart city concept being implemented in modern cities, the requirement of efficient ITS infrastructure, and scheduling algorithms to solve the challenges of latency, and energy efficiency are inevitable. Publisher-subscriber (pub-sub) model is an asynchronous, lightweight, and event-driven message communication system (B. Liu et al., 2021) that is most suitable for near real-time scenarios (e.g. vehicular networks) where dedicated communication is not guaranteed. The advent of 5G cellular networks has enabled higher throughput and V2X communication links; thus, the challenge of optimum scheduling of computational resources to process latency-sensitive tasks, in vehicular networks becomes more pertinent (Alvi et al., 2022; Rahman et al., 2022). The following are the important motivating points for undertaking this research work.

- Connected vehicle domain research promises, the solution to many of the challenges in the future smart cities with millions of connected vehicles.
- Advent of self-driving cars and high bandwidth communication systems, need for latency-sensitive services is growing. Thus, efficient resource scheduling and utilization, is of paramount importance, to match the ever-growing demand with limited computational resources in VFC.
- The QoS implementation for real-time, latency-sensitive vehicular applications is an open research challenge.
- Reducing energy consumption is an especially important research challenge in to march towards the green computing paradigm.
- The pub-sub communication model, an asynchronous, lightweight approach, can be utilized in conjunction with distributed VFC implementation to enhance QoS for real-time vehicular applications.

4.3 System Model Architecture

This work uses a three-layer system architecture consisting of the cloud, static fog, and vehicular fog layers as depicted in Fig 4.1. There is seamless connectivity in single or multiple hops among each of the vehicular nodes and VFC layer

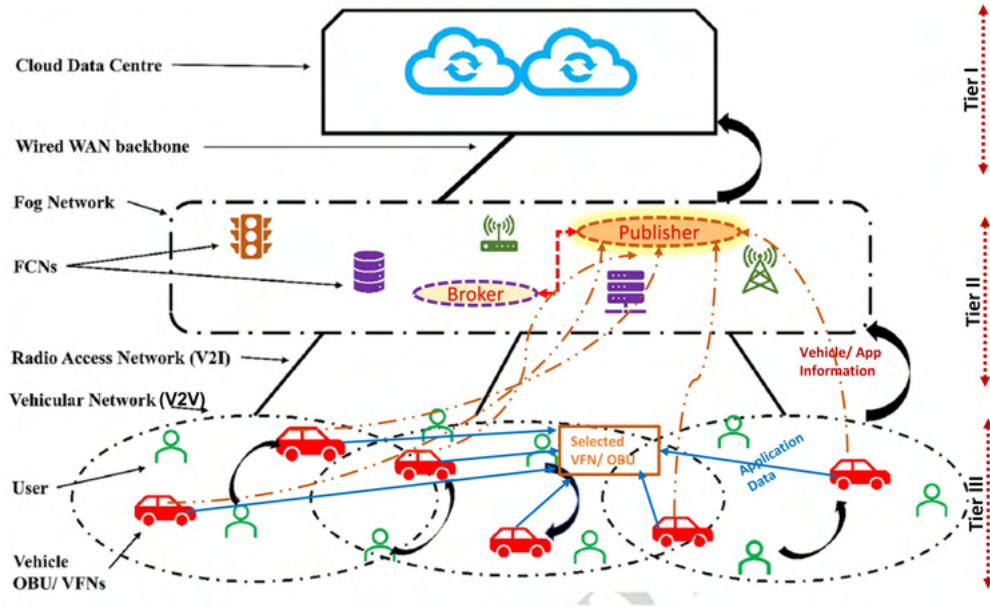


Figure 4.1 Three-tier VFC system architecture highlighting Pub-Sub model implementation

and static fog layer by radio links and between static fog and cloud layer using wired OFC-based connectivity. Pub-sub model is an event-driven distributed message dissemination system that works asynchronously and does not require direct/dedicated connectivity. There are three basic building blocks of this model namely: Publisher, Subscriber, and Message Broker as shown in Fig 4.1.

4.3.1 MOO Problem Formulation

The problem of selecting an optimal processing node (i.e., fog node) for a latency-sensitive request can be expressed as MOO model, aimed at minimizing overall energy while satisfying the maximum permissible latency for individual requests. Following is a mathematical expression for the proposed optimization problem formulation.

$$\text{Minimize } E = \sum_i \sum_j E(w_{ij}).Y_{ij}, \forall i \in v, j \in f \quad (4.1)$$

Subject to constraints:

$$\psi_i.\alpha \leq (1 - \theta).Q_i, \forall i \in f \quad (4.2)$$

$$\delta_{ij} \leq \delta_{max}, \forall i \in v, j \in f \quad (4.3)$$

$$Y_{ij} = \begin{cases} 1, & \text{if the pair } W_{ij} \text{ used for processing} \\ 0, & \text{otherwise} \end{cases} \quad (4.4)$$

$$0 \leq \theta \leq 1 \quad (4.5)$$

Equation (4.1) represents the objective function and equations (4.2) to (4.5) represent the associated constraints for the proposed model. Specifically, (4.2) represents the computational capacity conservation for the fog node, and (4.3) represents the maximum latency constraint for the request. In (4.4), Y_{ij} is a binary variable that represents whether a particular VFN-VN pair is being used for processing a request or not. Equation (4.5) represents a scaling variable θ , normalized processor occupancy factor. Q is total processing power of the node. Thus, $(1 - \theta).Q$ is the computational power that is available for utilization as a fog node.

4.4 Proposed Solution Technique for Optimal VFN Selection

The set of mathematical equations model energy efficiency as an optimal objective to select an optimal fog node f out of the pool of candidate fog nodes for any request r generated by vehicular node v . The energy efficiency optimization is subject to meeting the constraint of the QoS. Solving the given optimization problem using the conventional method is very time consuming, therefore unsuitable for providing a practical, and acceptable solution for scheduling real-time applications. Thus, an suitable technique design with reduced-complexity is needed. Driven by the above facts we present the LEAQoS algorithm following a heuristic approach.

4.4.1 Heuristics-based Algorithm for Solving the Mathematical Model

The Pareto technique used for multi-objective optimization has its limitations, particularly in that it can suffer from issues of slow convergence to the optimal front and inferior performance on problems with many objectives (M. Li, Yang, & Liu, 2016). Therefore, in this research work, we use the non-Pareto method (i.e., meta-heuristics) to receive optimal results for proposed multi-objective model. The LEAQoS algorithm proposed in this work uses meta-heuristic to achieve optimal results for two objectives of minimizing energy consumption and latency under the given constraints. The number of function calculations to denote the Pareto front can be huge and time-consuming, therefore non-dominated sorting technique (M. Li et al., 2016) is employed in this research work.

The OBU of different vehicular nodes in VFC has different hardware configurations. Every vehicular node may have a different set of intrinsic applications running on them, based on the make and configuration of the vehicle. Based on the hardware configuration and set of intrinsic applications running on it, different vehicular nodes will have different residual computational capacities, that can be utilized for vehicular fog computing. The complete research flow process is divided into three phases. First phase shortlists a small set of candidates VFN (CVFN) from a pool of all vehicular nodes, based on the available processor capability (measured as normalized processor load factor). In the second phase, the mapping of a request to the most suitable CVFN is performed. In the third phase of proposed solution strategy, total processing power of VFN system is adjusted according to instantaneous workload, by increasing or decreasing the number of CVFNs.

Table 4.2 Symbols used in Algorithms 1-3 and their description

Symbol	Description
n	Number of VN
N	Number of CVFN
VN[]	VN vector

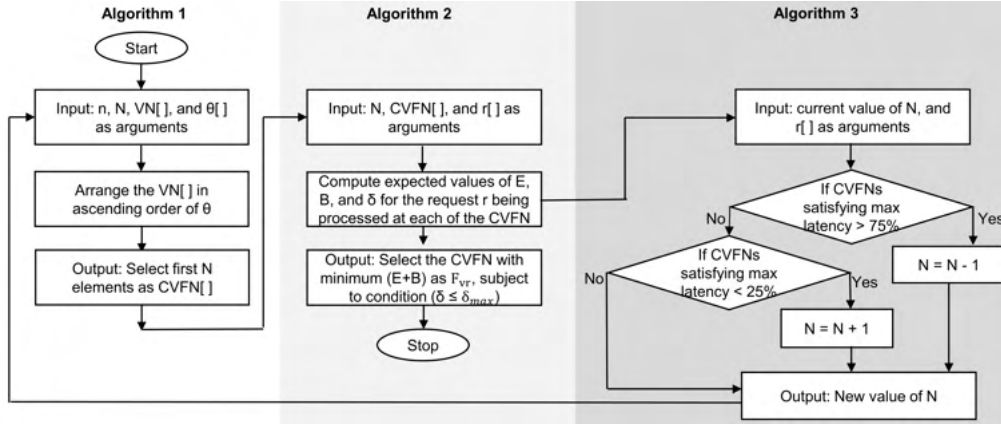


Figure 4.2 Flowchart of the proposed solution strategy indicating the role of each algorithm and their interdependence

θ	Normalized processor occupancy factor
CVFN[]	CVFN vector
δ	Latency
E	Energy
B	Network BW
δ_{max}	Maximum permissible latency
F_{vr}	Optimal VFN for processing of request r

Fig 4.2 depicts the complete solution approach proposed in this research work with the help of a flowchart for Algorithms 1, 2 and 3. Complete flowchart comprises of three algorithms as depicted in the flowchart. Table 4.2 gives symbols used in Algorithms 1 to 3 along with their description. The solution strategy for the selection of optimal fog node for a given latency-sensitive request is divided into three phases. Each phase is implemented as an algorithm, as depicted clearly in Fig 4.2.

The first phase (i.e., CVFN selection) involves shortlisting the available VNs based on the available processing capability. The publisher maintains and publishes the updated information database (i.e., position, direction, speed, communication interfaces, available processing capacity, etc.) about each VN in the network. Algorithm 1 starts by taking all the input arguments that are pushed

by the publisher to subscribers via the message broker in the proposed pub-sub model (N is assigned an initial value of 4). The VNs are arranged in the ascending order of normalized processor occupancy factor. At the end of Algorithm 1, N number of VNs with the lowest normalized processor occupancy factor are selected as CVFN. The CVFN vector from the output of Algorithm 1 is used as the input to Algorithm 2, along with the request to be processed.

Algorithm 1 Selection of candidate vehicular fog nodes (CVFN)

```

1: Inputs: Number of vehicular nodes VN ( $n$ ), Number of candidate vehicular fog nodes CVFN ( $N$ ),
2:           Vehicular node vector (VN), Normalized processor occupancy factor of VN ( $\theta$ )
3: Output: Candidate vehicular fog node vector (CVFN [ ])
4: Procedure: SELECT CVFN [N]
5:   for the number of vehicular nodes from 1 to  $n$ 
6:     arrange VN as per ascending order of  $\theta$ 
7:   end for
8:   for  $i = 1$  to  $N$ 
9:     CVFN [ $i$ ]  $\leftarrow$  VN
10:  end for
11: End procedure
12: Return CVFN [ ]

```

In the second phase, Algorithm 2 is executed each time a new application request is received for processing. Whenever Algorithm 2 executes, it computes the expected values of the energy consumption, bandwidth consumption, and latency for the presented application request, from each of the CVFNs as per the MOO mathematical model. From this step of Algorithm 2 the values of latency in respect of each CVFN are passed onto Algorithm 3 as depicted in Fig 4.2. From earlier step output value, optimal fog node is the CVFN with a minimum value of total resource (energy + bandwidth) consumption, subject to the satisfaction of the maximum latency limit for the application request.

Algorithm 2 Lightweight Energy Efficient Algorithm for QoS implementation in VFC (LEAQoS)

```
1: Inputs: Number of candidate VFN ( $N$ ), candidate VFN vector ( $CVFN[ ]$ ),  
2:       New request ( $r$ ), Maximum permissible latency for  $r$  ( $\delta_{max}$ ).  
3: Output: Optimal fog node for processing of the request ( $F_{vr}$ ).  
4: Procedure: SELECT FOG NODE ( $F_{vr}$ ).  
5: for number of requests from 1 to  $R$   
6:   for number of CVFN from 1 to  $N$   
7:     find network bandwidth consumption  $B[ ]$ ;  
8:     find energy consumption  $E[ ]$ ;  
9:     find latency  $\delta[ ]$ ;  
10:     $T[ ] \leftarrow E[ ] + B[ ]$ ; (total resource consumption)  
11:   end for  
12: Arrange  $CVFN[ ]$  in ascending order of  $T[ ]$ ;  
13:   for  $i = 1$  to  $N$   
14:     if ( $\delta[i] \leq \delta_{max}$ )  
15:        $F_{vr} \leftarrow CVFN[i]$ ;  
16:     end if  
17:   end for  
18: end for  
19: End procedure  
20: Return ( $F_{vr}$ )
```

In the third phase, Algorithm 3 executes to adaptively regulate the processing capacity of the VFC system so that the latency and resource consumption are optimized. The execution of Algorithm 3 is triggered as a follow-up action after the execution of Algorithm 2 (end of the second step). If number of CVFNs satisfying the maximum latency constraint (for the current application request being processed) is within 25 to 75 percent, no action is needed to adjust the capacity of the VFC system. Conversely, if number of the CVFN satisfying maximum latency constraint is less than 25 percent or more than 75 percent, number of CVFN (i.e., capacity of VFC) is to be increased or decreased by one, respectively. In this work, the values of 75 percent and 25 percent, represent the probability of processing the request at VFN and are chosen empirically. Algorithm 3 estimates the processing capacity of the VFC system for forthcoming application requests. Thus, ensuring that the optimal capacity of the VFC is always maintained, resulting in maximum latency satisfaction for latency-sensitive requests and, minimum resource consumption. The latest value of the number of CVFN from the output of Algorithm 3 is taken as input by Algorithm 1. In the flowchart, in Fig 4.2, Algorithm 2 execution is triggered by an incoming application request. Algorithm 3 is triggered by the output of the second step of Algorithm 2. Algorithm 1 is executed at the end of Algorithm 3 or at a predetermined interval (defined as 10 ms in the simulation), whichever is earlier.

Algorithm 3 Adaptive Capacity Tuning for Dynamic Workload (ACT-DW)

```
1: Inputs: Current value of number of CVFN (N), New request vector (r,  $\delta_{max}$ )
2: Output: New value of number of CVFN (N).
3: Procedure: FIND OPTIMAL NUMBER OF CVFN (N).
4: for number of request from 1 to R
5:     k = 0;
6:     for number of CVFN, i from 1 to N
7:         if ( $\delta[i] \leq \delta_{max}$ ) (as per Algorithm 2)
8:             k ++ ;
9:         end if
10:    end for
11:    if ( $k > 0.75 * N$ )
12:        N --;
13:    else if ( $k < 0.25 * N$ )
14:        N ++;
15:    else
16:        N = N;
17:    end for
18: End procedure.
19: Return (N) ;
```

Algorithm-1 selects the required number of the CVFN out of the available VNs, based on the availability of surplus computational power with the VN at the current instance. In this research work, an urban scenario is assumed, where all the VNs have adequate wireless communication coverage for V2V and V2I communication. Different VNs have different hardware configurations and different sets of intrinsic applications running on them. Besides, some of the VNs may have already been selected as VFNs, and currently processing a latency-sensitive request. Above mentioned factors decide the current processor occupancy factor for the VN. The inverse of the normalized processor occupancy factor is the measure of the surplus computational capacity available with the VN. In VFC the nodes with surplus computational capacity are selected as VFNs for processing latency-sensitive requests. Therefore, the normalized processor occupancy factor for the VN is taken as a reference for selecting the CVFN.

Phase two of the proposed solution uses a heuristics approach to select optimal fog nodes out of the pool of CVFNs. Pseudo-code of proposed lightweight energy-efficient algorithm for QoS implementation (LEAQoS) is represented in

Algorithm 2 in Fig 4.2 and detailed steps are provided in Annexure II. Algorithm 2 takes the candidate VFN vector (CVFN []) from Algorithm 1, and the request vector [r] as inputs (steps 1 and 2) and returns the optimal fog node (F_{vr}) for processing the request (step 3). Steps 6 to 9 of the algorithm compute the values of network bandwidth consumption, energy consumption, and latency for processing the request at each of the CVFNs. Step 10 computes the cumulative value of resource consumption T for each of the requests to be processed. The resultant resource consumption vector T [] is arranged in ascending order in step 12. In steps 13 to 17, the assignment decision for the request is made, to satisfy the maximum permissible latency value for the request.

Algorithm 2 is very lightweight in computational resources since the computational load is only limited to computing very few latency, network BW, and energy consumption values (typically in the range of 3 to 6) for each latency-sensitive request. LEAQoS assigns the request to optimal VFN for processing, considering QoS (given by latency, energy consumption, and network bandwidth) satisfaction for the request.

To satisfy QoS for dynamic workload, the resource availability in VFC must adapt dynamically to the workload. In phase three of this research work, the computational capacity of VFC (i.e., size of the pool of candidate vehicular fog nodes) is dynamically adjusted as per instantaneous workload.

The Pseudo code for ACT-DW in VFC is depicted in Algorithm 3 in Fig 4.2 and detailed steps are provided in Annexure III. This algorithm takes inputs as latency vectors for all current requests being processed in VFN, the current value of the number of CVFN (i.e., N), and the current request vector [r], from Algorithm-2 and, returns a new value of the number of CVFN. In steps 5 through 10, the algorithm computes the processing factor (k) at VFN for current requests by a cumulative count of several instances wherein, the achieved latency is less than or equal to the maximum permissible latency for a request. The algorithm estimates the instantaneous availability of resources at VFN for the processing of the next request. If the probability of processing at VFN is more than 75%, it indicates that surplus VFN resources are available. Accord-

ingly, the number of CVFNs is reduced by one. Conversely, if the probability of processing at VFN is less than 25%, it indicates a shortage of VFN resources for meeting QoS. Hence, the number of CVFNs is increased by one. In case the probability of processing at VFN is between 25% and 75%, no change in VFN capacity is made (steps 11 through 16). Thus, the algorithm adaptively adjusts the availability of VFN resources by increasing or reducing number of CVFNs, as per current workload and processing power availability. Output of Algorithm-3 is used by Algorithm-1, recursively for selecting the pool of CVFN.

4.5 Important Contributions

This research work proposes an energy-efficient technique, to address the issue of allocating dynamically generated requests, to dynamically available VFN resources. Specifically, our proposed algorithm(s) solves the problem of optimization of energy efficiency for vehicular networks, while meeting the QoS demand of individual application requests. Additionally, this research work proposes a novel concept of computing offloading by using a publisher-subscriber (pub-sub) based approach, to dynamically select the VFN out of available vehicular nodes (VN). In computing offloading the computational workload (i.e., control plane processing) for the selection of optimal VFN for a given request, is offloaded to the static fog node instead of processing it at VFN. This unique approach is nature-inspired. Nature has abundant systems wherein, the tasks are equitably distributed for better management and efficiency, e.g., the functioning of the human body under the control of the human brain. This ensures that the presented technique uses minimum computing resources. Solution strategy proposed is a heuristics-based algorithm to assign the dynamically generated, latency-sensitive request to the optimal fog node. Finally, this work proposes an algorithm for adaptively adjusting the VFC resources as per the instantaneous workload, so that excess VFN resources are not active at any given time, to minimize the latency, and cost to the fog service provider. Important outcomes of this research work are enumerated below.

- A novel algorithm is presented for resource optimization in VFC while sat-

isfying the maximum permissible latency parameter for real-time, latency-sensitive requests.

- Mathematical modeling of the proposed technique as a MOO model and a heuristic-based LEAQoS.
- An adaptive capacity tuning algorithm for the dynamic workload (ACTDW) is presented, to adaptively adjust the real-time availability of VFC resources, based on the instantaneous workload, thus, minimizing the latency, and cost to the VFC service provider.
- A unique, nature-inspired pub-sub model is proposed for computing offloading, with algorithm processing at FCN, which ensures minimum compute overhead at VFN. Thus, minimizing network bandwidth consumption, and maximum processing power of VFNs is available for processing latency-sensitive requests, ensuring maximum QoS satisfaction.

4.6 Summary

Further to the VFC system design and mathematical model presented in Chapter 3, in this chapter the novel implementation of the well-known publisher-subscriber communication architecture is presented. The MOO model is established to minimize energy consumption, with condition of fulfilling maximum permissible latency for the latency-sensitive request. The problem of solving the MOO model is proposed in the form of heuristic-based approach. Solution consists of three algorithms. Complete solution technique to the problem of selecting the optimal fog node for processing a real-time request is proposed. The solution presented also takes care of dynamically generated workload along with dynamic resource availability by presenting the ACT-DW algorithm. Presented methodology is an iterative process that repeats with every incoming latency-sensitive application request.

Chapter 5

Performance Evaluation

5.1 Introduction

This chapter presents the details of the simulation performance evaluation for the proposed research work. The Section 5.2 elaborates the simulation setup. Section 5.3 presents the detailed results of simulation and discussion on the results analysis. Finally, Section 5.4 presents the summary of the chapter.

Every VN is heterogeneous in terms of its resource capability, instantaneous communication capability, and workload request generation profile. The workload requests generated by VN are mutually exclusive. More than one request can be generated by a VN simultaneously and, they may be processed at different fog nodes. It is assumed that 65% of total requests generated are latency-sensitive and the highest value of 250 ms is the threshold that defines the request as a latency-sensitive request (Mahmud et al., 2018). realistic vehicular traffic data generated from Chandigarh city; India is used for simulation. As depicted in the simulation setup in Fig 5.1, SUMO (Behrisch et al., n.d.), is used for generating this traffic dataset. Input to SUMO is a map (3.5 x 3.0 km) of Chandigarh city, India, from the OSM, as depicted in Fig 5.2. The output FCD (floating car data) file (Behrisch et al., n.d.) contains location and speed, vehicle angle, position, lane information, etc. at every predefined time period. OMNET++ (Sommer et al., 2019) has been used for simulating V2X communication, and iFogsim2 (Mahmud, Pallewatta, Goudarzi, & Buyya, 2021) is used for simulating the application processing in a fog environment. TraCI (Traffic Control interface)

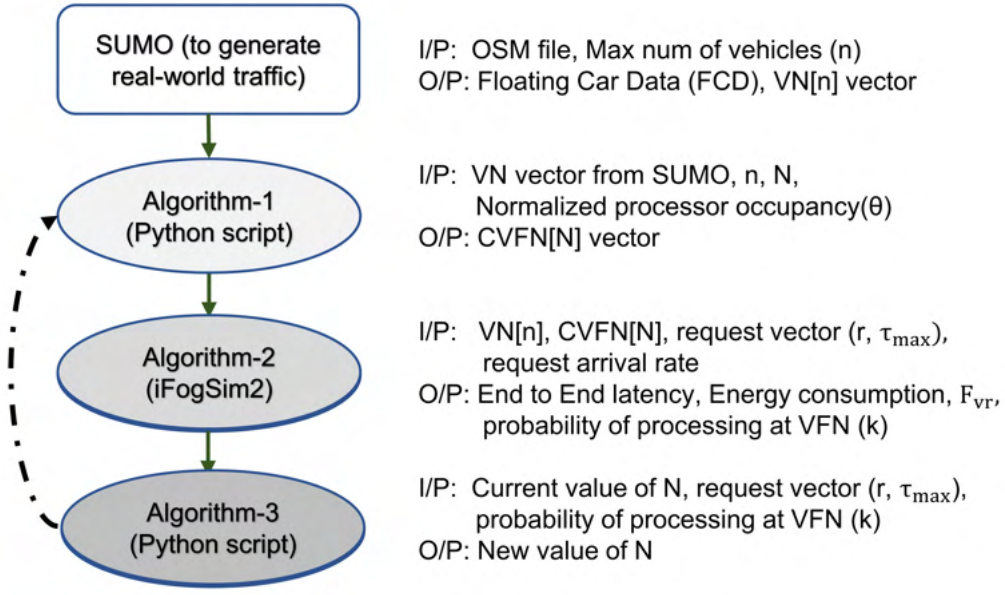


Figure 5.1 Performance evaluation setup and flow of simulations

provides interaction between SUMO, and OMNET++, iFogSim2. Using the ‘traceExporter.py’ python utility the SUMO FCD file is filtered to create different trace file formats for use as input to OMNET++ and iFogSim2 for simulation and analysis.

The proposed optimal VFN selection technique for the latency-sensitive requests is evaluated for different decision variables e.g., number of vehicles, request generation rate, maximum permissible latency, and request sizes. Simulation run are compared to (Alvi et al., 2022; Yadav et al., 2020; Hameed et al., 2021) (under urban road conditions) to study the relative quantitative QoS advantage of the proposed technique.

5.2 Simulation Environment

A well-known tool, SUMO (simulation of urban mobility) (Behrisch et al., n.d.), is used for generating realistic vehicle parameters. However, the application request parameters are generated synthetically, as no real-world data/ simulators are currently available for generating such application requests. A normalized processor occupancy factor value (percentage) in the range of 25 - 60 is used.

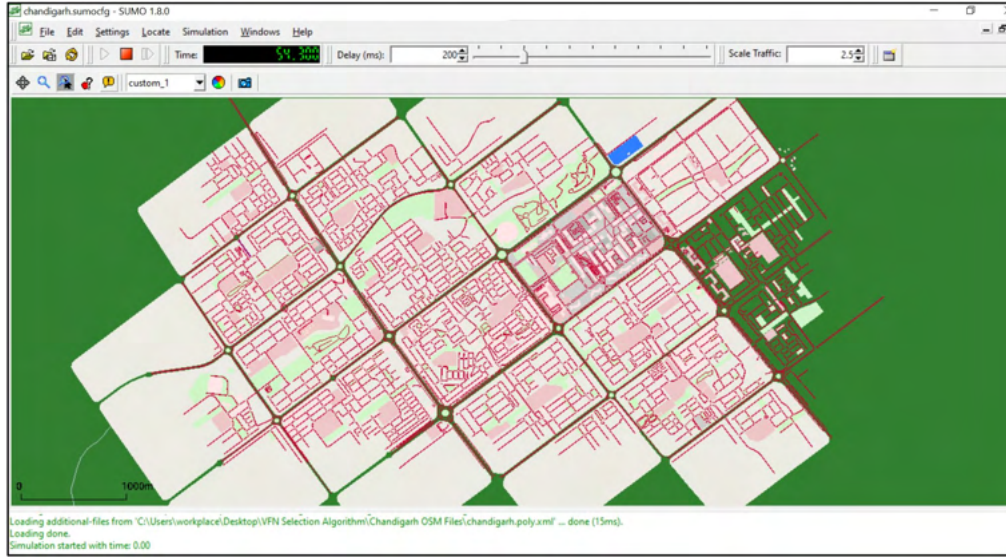


Figure 5.2 Snapshot of the SUMO while running the simulation setup

The specific values of the simulation parameter in a given range are computed by the Pseudorandom number generator function in Python (i.e., `randint()` function). The SUMO takes the map of Chandigarh city, India, from the open street map (using OSM web wizard tool in SUMO)(Behrisch et al., n.d.), with an area of 3.5 x 3.0 km as input for generating realistic urban traffic. The floating car data (FCD) file (Behrisch et al., n.d.), obtained from SUMO is used to generate vehicular node vectors containing different vehicle parameters. A snapshot of the SUMO running the simulation using the Chandigarh city map (obtained from the open street map) is depicted in Fig 5.2. FCD is an output trace file from SUMO that contains location, speed, vehicle angle, position, and lane along with other data about each vehicle in the network, at each time step. A snapshot of the FCD data export (*.xml) from SUMO is presented in Fig 5.3.

Python code is used for implementing Algorithm-1. Inputs to Algorithm-1 are the output of the SUMO FCD file, normalized processor occupancy factor vector, and initialized values of several VN and numbers of CVFN. Algorithm 1 generates the CVFN vector. For simulating Algorithm-2, a well-known fog simulation tool iFogSim2 (Mahmud et al., 2021), which is an extension of iFogSim (Gupta, Vahid Dastjerdi, Ghosh, & Buyya, 2017), and built upon the CloudSim (Calheiros, Ranjan, Beloglazov, De Rose, & Buyya, 2011) platform is used. iFogSim is a widely used simulation platform for simulating different fog com-

```

<fcd-export>
<vehicle id="veh0" x="76.771212" y="30.734011" angle="309.627872" type="veh_passenger" speed="22.691739" pos="55.691178" lane="12954496285 0" slope="0.000000"/>
<vehicle id="veh1" x="76.794755" y="30.739245" angle="141.212558" type="veh_passenger" speed="11.866810" pos="2.624228" lane="12972074584 0" slope="0.000000"/>
<vehicle id="veh10" x="76.773758" y="30.739931" angle="131.275770" type="veh_passenger" speed="16.385756" pos="36.170179" lane="12972074584 0" slope="0.000000"/>
<vehicle id="veh11" x="76.787521" y="30.747628" angle="323.251989" type="veh_passenger" speed="14.311794" pos="50.141113" lane="12972074484 0" slope="0.000000"/>
<vehicle id="veh12" x="76.793923" y="30.749862" angle="323.487944" type="veh_passenger" speed="25.677591" pos="48.127443" lane="12978239581 0" slope="0.000000"/>
<vehicle id="veh13" x="76.762393" y="30.740731" angle="353.081096" type="veh_passenger" speed="13.463220" pos="129.858863" lane="12955349683 0" slope="0.000000"/>
<vehicle id="veh14" x="76.795404" y="30.740504" angle="323.206651" type="veh_passenger" speed="15.276503" pos="144.615698" lane="12972075387 0" slope="0.000000"/>
<vehicle id="veh15" x="76.796253" y="30.739858" angle="142.847354" type="veh_passenger" speed="8.671817" pos="6.067055" lane="1430553175 30" slope="0.000000"/>
<vehicle id="veh16" x="76.762561" y="30.748781" angle="142.541940" type="veh_passenger" speed="19.824555" pos="33.160783" lane="12847705990 0" slope="0.000000"/>
<vehicle id="veh17" x="76.773240" y="30.729540" angle="57.521232" type="veh_passenger" speed="18.138544" pos="36.724221" lane="1295222972 0" slope="0.000000"/>
<vehicle id="veh18" x="76.788011" y="30.733070" angle="330.227277" type="veh_passenger" speed="17.772136" pos="29.848693" lane="12950447484 0" slope="0.000000"/>
<vehicle id="veh19" x="76.797884" y="30.752897" angle="204.205430" type="veh_passenger" speed="12.797018" pos="72.489743" lane="12972294985 0" slope="0.000000"/>
<vehicle id="veh20" x="76.786990" y="30.731474" angle="50.812339" type="veh_passenger" speed="17.836989" pos="78.615592" lane="12954354683 0" slope="0.000000"/>
<vehicle id="veh21" x="76.784991" y="30.727399" angle="236.028558" type="veh_passenger" speed="15.427840" pos="2.315657" lane="1429141813 0" slope="0.000000"/>
<vehicle id="veh22" x="76.800365" y="30.736773" angle="321.831842" type="veh_passenger" speed="22.114371" pos="0.089548" lane="12847705990 0" slope="0.000000"/>
<vehicle id="veh23" x="76.762874" y="30.744417" angle="142.541940" type="veh_passenger" speed="18.452166" pos="83.476531" lane="12847705990 0" slope="0.000000"/>
<vehicle id="veh24" x="76.764011" y="30.741267" angle="145.714911" type="veh_passenger" speed="7.053501" pos="3.664324" lane="12953785140 0" slope="0.000000"/>
<vehicle id="veh25" x="76.770143" y="30.734210" angle="44.827941" type="veh_passenger" speed="18.496705" pos="29.392810" lane="129545037815 0" slope="0.000000"/>
<vehicle id="veh26" x="76.789741" y="30.746574" angle="318.805221" type="veh_passenger" speed="18.252310" pos="102.316549" lane="12972994862 0" slope="0.000000"/>
<vehicle id="veh27" x="76.800923" y="30.729243" angle="322.909311" type="veh_passenger" speed="12.575058" pos="122.380424" lane="1295346682 0" slope="0.000000"/>
<vehicle id="veh28" x="76.810175" y="30.724574" angle="317.931911" type="veh_passenger" speed="26.283096" pos="286.984072" lane="12950895982 0" slope="0.000000"/>
<vehicle id="veh29" x="76.767082" y="30.729668" angle="322.833754" type="veh_passenger" speed="13.103022" pos="156.784574" lane="12953462148 0" slope="0.000000"/>
<vehicle id="veh3" x="76.773040" y="30.748152" angle="232.493045" type="veh_passenger" speed="29.307435" pos="144.692226" lane="12977732884 0" slope="0.000000"/>
<vehicle id="veh30" x="76.805203" y="30.727280" angle="241.189598" type="veh_passenger" speed="18.474601" pos="105.578125" lane="12958099892 0" slope="0.000000"/>
<vehicle id="veh31" x="76.760882" y="30.747242" angle="51.711593" type="veh_passenger" speed="17.626827" pos="40.108718" lane="12977732884 0" slope="0.000000"/>
<vehicle id="veh32" x="76.798756" y="30.744801" angle="323.082921" type="veh_passenger" speed="16.981743" pos="156.827711" lane="12972072547 0" slope="0.000000"/>
<vehicle id="veh33" x="76.800279" y="30.732107" angle="320.326928" type="veh_passenger" speed="12.905481" pos="111.021403" lane="1299434284 0" slope="0.000000"/>
<vehicle id="veh34" x="76.788081" y="30.762509" angle="53.362708" type="veh_passenger" speed="9.144855" pos="23.001480" lane="12978472148 0" slope="0.000000"/>
<vehicle id="veh35" x="76.799154" y="30.731890" angle="232.733179" type="veh_passenger" speed="8.781988" pos="89.970717" lane="12958583940 0" slope="0.000000"/>
<vehicle id="veh36" x="76.786556" y="30.737761" angle="54.210613" type="veh_passenger" speed="5.208351" pos="3.859674" lane="12954499087 0" slope="0.000000"/>
<vehicle id="veh37" x="76.781837" y="30.730157" angle="53.364655" type="veh_passenger" speed="12.688329" pos="10.565811" lane="12954349683 0" slope="0.000000"/>
<vehicle id="veh38" x="76.789334" y="30.751793" angle="143.228629" type="veh_passenger" speed="16.055267" pos="96.864707" lane="129784731 0" slope="0.000000"/>
<vehicle id="veh39" x="76.806625" y="30.742608" angle="143.081713" type="veh_passenger" speed="14.694413" pos="69.620565" lane="12972072548 0" slope="0.000000"/>
<vehicle id="veh4" x="76.763547" y="30.739188" angle="323.878656" type="veh_passenger" speed="11.521389" pos="11.671852" lane="1294644984 0" slope="0.000000"/>
<vehicle id="veh40" x="76.763946" y="30.737801" angle="51.896261" type="veh_passenger" speed="11.677924" pos="103.855157" lane="12953776684 0" slope="0.000000"/>
<vehicle id="veh41" x="76.764554" y="30.740651" angle="322.523960" type="veh_passenger" speed="6.238628" pos="4.362720" lane="12953785140 0" slope="0.000000"/>
<vehicle id="veh42" x="76.787823" y="30.737079" angle="53.364655" type="veh_passenger" speed="13.743146" pos="129.858863" lane="12955349683 0" slope="0.000000"/>
<vehicle id="veh43" x="76.764685" y="30.742723" angle="233.317014" type="veh_passenger" speed="7.991840" pos="13.356753" lane="12954499783 0" slope="0.000000"/>
<vehicle id="veh102" x="76.778816" y="30.732902" angle="53.153867" type="veh_passenger" speed="0.000000" pos="31.826683" lane="12946403987 0" slope="0.000000"/>
<vehicle id="veh109" x="76.779429" y="30.733635" angle="37.394577" type="veh_passenger" speed="0.000000" pos="10.915648" lane="12954507583 0" slope="0.000000"/>
<vehicle id="veh11" x="76.779488" y="30.733156" angle="246.964291" type="veh_passenger" speed="0.000000" pos="4.637529" lane="1429078403 0" slope="0.000000"/>
<vehicle id="veh110" x="76.779488" y="30.733156" angle="246.964291" type="veh_passenger" speed="0.000000" pos="4.637529" lane="1429078403 0" slope="0.000000"/>
<vehicle id="veh111" x="76.779594" y="30.733293" angle="191.386773" type="veh_passenger" speed="0.000000" pos="3.834363" lane="1429078415 0" slope="0.000000"/>
<vehicle id="veh113" x="76.778597" y="30.733498" angle="76.658886" type="veh_passenger" speed="0.000000" pos="4.487738" lane="1429078519 0" slope="0.000000"/>
<vehicle id="veh118" x="76.779574" y="30.733231" angle="201.014788" type="veh_passenger" speed="0.000000" pos="6.345422" lane="12954507587 0" slope="0.000000"/>
<vehicle id="veh140" x="76.778404" y="30.733451" angle="7.688259" type="veh_passenger" speed="0.000000" pos="7.928565" lane="12954507584 0" slope="0.000000"/>
<vehicle id="veh145" x="76.779415" y="30.733516" angle="7.601303" type="veh_passenger" speed="0.000000" pos="0.273615" lane="1429078473 0" slope="0.000000"/>
<vehicle id="veh146" x="76.779470" y="30.733212" angle="314.705420" type="veh_passenger" speed="0.000000" pos="5.532990" lane="12954507581 0" slope="0.000000"/>
<vehicle id="veh150" x="76.779679" y="30.733155" angle="267.758287" type="veh_passenger" speed="0.000000" pos="6.839873" lane="12954507580 0" slope="0.000000"/>
<vehicle id="veh160" x="76.778530" y="30.733171" angle="248.963615" type="veh_passenger" speed="0.000000" pos="7.621910" lane="1429078407 0" slope="0.000000"/>
<vehicle id="veh161" x="76.778749" y="30.732822" angle="53.193867" type="veh_passenger" speed="0.000000" pos="16.824509" lane="12946403987 0" slope="0.000000"/>
<vehicle id="veh162" x="76.780059" y="30.733553" angle="156.512983" type="veh_passenger" speed="0.000000" pos="5.060325" lane="1429078454 1" slope="0.000000"/>
<vehicle id="veh168" x="76.778485" y="30.732193" angle="53.193867" type="veh_passenger" speed="0.000000" pos="9.323449" lane="12946403987 0" slope="0.000000"/>
<vehicle id="veh173" x="76.778763" y="30.733139" angle="92.665617" type="veh_passenger" speed="0.000000" pos="4.130077" lane="12954507584 1" slope="0.000000"/>
<vehicle id="veh184" x="76.779915" y="30.733649" angle="101.194672" type="veh_passenger" speed="0.000000" pos="5.638176" lane="1429078507 0" slope="0.000000"/>
<vehicle id="veh186" x="76.761590" y="30.755852" angle="326.140562" type="veh_passenger" speed="22.431392" pos="118.206972" lane="12947691482 0" slope="0.000000"/>
<vehicle id="veh193" x="76.778464" y="30.733486" angle="175.679416" type="veh_passenger" speed="0.000000" pos="12.561326" lane="1429078454 1" slope="0.000000"/>
<vehicle id="veh198" x="76.778423" y="30.733267" angle="322.548478" type="veh_passenger" speed="0.000000" pos="6.64062" lane="1429078415 0" slope="0.000000"/>
<vehicle id="veh199" x="76.778539" y="30.732941" angle="53.193867" type="veh_passenger" speed="0.000000" pos="39.327691" lane="12946403987 0" slope="0.000000"/>
<vehicle id="veh204" x="76.778464" y="30.733265" angle="248.190098" type="veh_passenger" speed="0.000000" pos="18.637710" lane="12954507580 1" slope="0.000000"/>
<vehicle id="veh207" x="76.779607" y="30.733168" angle="281.979720" type="veh_passenger" speed="0.000000" pos="0.120877" lane="1429078407 0" slope="0.000000"/>
<vehicle id="veh208" x="76.761390" y="30.756181" angle="326.140562" type="veh_passenger" speed="22.386987" pos="153.349880" lane="12947691482 0" slope="0.000000"/>
<vehicle id="veh209" x="76.780013" y="30.733557" angle="188.078360" type="veh_passenger" speed="0.000000" pos="14.533341" lane="12954507586 0" slope="0.000000"/>
<vehicle id="veh211" x="76.778464" y="30.733270" angle="53.217131" type="veh_passenger" speed="0.000000" pos="31.711991" lane="12946403983 0" slope="0.000000"/>
<vehicle id="veh214" x="76.779468" y="30.733802" angle="123.623788" type="veh_passenger" speed="0.000000" pos="26.075890" lane="12954513883 0" slope="0.000000"/>
<vehicle id="veh215" x="76.779904" y="30.733062" angle="307.278527" type="veh_passenger" speed="0.000000" pos="33.304681" lane="5743049864 0" slope="0.000000"/>
<vehicle id="veh218" x="76.779771" y="30.733488" angle="95.586166" type="veh_passenger" speed="0.000000" pos="6.616106" lane="12954507584 0" slope="0.000000"/>
<vehicle id="veh219" x="76.778554" y="30.733337" angle="31.292450" type="veh_passenger" speed="0.000000" pos="0.076027" lane="1429078415 1" slope="0.000000"/>
<vehicle id="veh220" x="76.779227" y="30.733166" angle="37.846434" type="veh_passenger" speed="0.000000" pos="12.132911" lane="12946403983 0" slope="0.000000"/>
<vehicle id="veh221" x="76.780057" y="30.733420" angle="172.514433" type="veh_passenger" speed="0.000000" pos="7.086752" lane="12954507586 1" slope="0.000000"/>
<vehicle id="veh222" x="76.779768" y="30.733127" angle="296.190478" type="veh_passenger" speed="0.000000" pos="5.066704" lane="1429078403 1" slope="0.000000"/>
<vehicle id="veh225" x="76.779603" y="30.732961" angle="53.193867" type="veh_passenger" speed="0.000000" pos="46.828697" lane="12946403987 0" slope="0.000000"/>
<vehicle id="veh228" x="76.779313" y="30.733279" angle="30.246556" type="veh_passenger" speed="0.000000" pos="27.134967" lane="12946403983 0" slope="0.000000"/>
<vehicle id="veh232" x="76.778937" y="30.733096" angle="298.460831" type="veh_passenger" speed="0.000000" pos="40.895665" lane="5743049864 0" slope="0.000000"/>
<vehicle id="veh234" x="76.779521" y="30.733467" angle="327.733467" type="veh_passenger" speed="0.000000" pos="10.771601" lane="1429078415 0" slope="0.000000"/>
<vehicle id="veh252" x="76.771029" y="30.754746" angle="323.630904" type="veh_passenger" speed="13.009963" pos="42.513977" lane="129799633 0" slope="0.000000"/>
<vehicle id="veh256" x="76.780648" y="30.733489" angle="195.547965" type="veh_passenger" speed="0.000000" pos="6.605730" lane="1429078464 0" slope="0.000000"/>
<vehicle id="veh259" x="76.778178" y="30.733132" angle="37.838353" type="veh_passenger" speed="0.000000" pos="4.621853" lane="12946403983 0" slope="0.000000"/>
<vehicle id="veh263" x="76.779666" y="30.733021" angle="53.221580" type="veh_passenger" speed="0.000000" pos="54.329784" lane="12946403987 0" slope="0.000000"/>
<vehicle id="veh264" x="76.778176" y="30.734023" angle="160.689125" type="veh_passenger" speed="0.000000" pos="10.270603" lane="1431038882 0" slope="0.000000"/>
<vehicle id="veh267" x="76.780027" y="30.732978" angle="319.927159" type="veh_passenger" speed="0.000000" pos="18.302643" lane="5743049864 0" slope="0.000000"/>
<vehicle id="veh272" x="76.779844" y="30.733029" angle="307.278527" type="veh_passenger" speed="0.000000" pos="28.804684" lane="5743049864 0" slope="0.000000"/>
<vehicle id="veh533" x="76.780684" y="30.739533" angle="515.684265" type="veh_passenger" speed="0.000000" pos="10.801631" lane="6743049864 0" slope="0.000000"/>

```

Figure 5.3 Snapshot of the FCD export file from SUMO during the simulation run

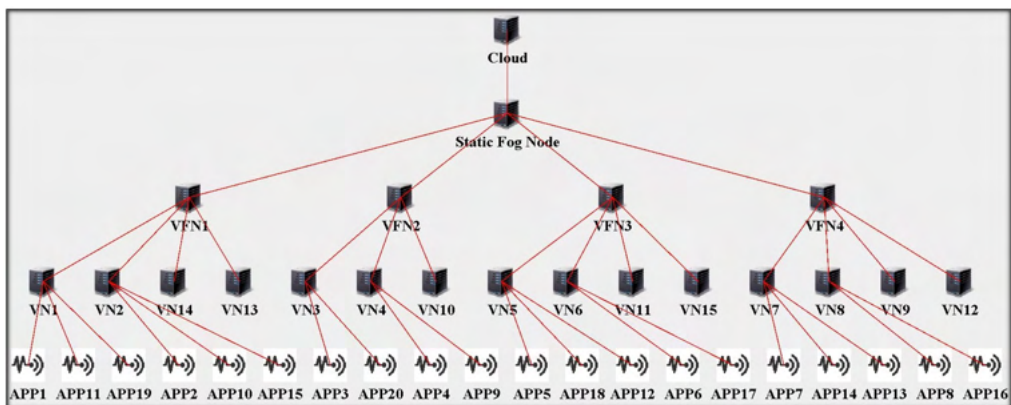


Figure 5.4 Snapshot of the simulation setup in iFogSim2

puting paradigms. Fig 5.4 depicts a snapshot of the simulation topology running in iFogSim2. Values of various simulation parameters used are given in Table 5.1 and values of various parameters of request vector are given in Table 5.2 are adopted from (Mahmud et al., 2018; Yadav et al., 2020; Hameed et al., 2021; Mahmud et al., 2021; Z. Wang, Zheng, Ge, & Li, 2020). Table 5.3 represents different problem configurations used during the simulation. Algorithm-3 is implemented using Python script. The inputs are the current value of the number of CVFN, the current request vector, and the corresponding probability of processing at VFN. This algorithm estimates the number of VFNs required as per instantaneous workload and processing resources. Algorithm 3 tunes the VFN resources dynamically so that not more than current resource requirements are held up, simultaneously scarcity of computing power is also met. This algorithm could be very useful for optimizing billing /pricing, based on the usage of VFN resources and implementing service level agreements (SLA) (Binwal & Kapoor, 2022), to implement a pay-per-use model in VFC.

Table 5.1 Values of various simulation parameters

Parameter Name	Values
The area of the road considered	3.5 x 3.0 km
Uplink bandwidth (VN to VFN)	10 Mbps
Uplink bandwidth (VFN to FCN)	10 Mbps
Uplink bandwidth (FCN to Cloud)	100 Mbps
Number of VN (i.e., vehicles)	20 - 100
Power consumption VFN (busy/idle)	87 / 82 J
Power consumption FCN (busy/idle)	107 / 83 J
Power consumption Cloud (busy/idle)	1648 / 1328 J
Number of VFNs	3 - 6
Vehicle speed	0 - 40 kmph
Max communication radius of VNs	300 meters
Simulation duration	250 sec
Uplink latency (VN to VFN)	2 ms
Uplink latency (VFN to FCN)	74 ms
Uplink latency (FCN to Cloud)	100 ms

For performance evaluation, five different problem configurations have been chosen, and a comparative study of present technique with (Alvi et al., 2022; Yadav et al., 2020), and (Hameed et al., 2021) (under urban road conditions) is presented. Table 5.3 depicts the different problem configurations used for performance evaluation and comparison. Varying number of VNs means corresponding variation in request arrival rate, thus the result parameters also vary. The number of vehicles indicated against each simulation configuration is the maximum number of VNs active in the simulation. Results have been plotted by averaging the results of five simulation runs by choosing random values from requests per second ranges in Table 5.3. Tables 5.4, 5.5, and 5.6 present the standard deviation of obtained values for each configuration simulation run, for latency, energy consumption, and network usage, respectively. A state-based energy consumption model was used during the simulation. The power consumption values are off power : 0 mW, idle power: 2 mW, receiver busy power: 5 mW, receiver receiving power: 10 mW, sleep power: 1 mW, transmitter transmitting power: 100 mW.

Table 5.2 Values of various parameters of request vector

Parameter Name	Value
Size of request	100 – 300 Kbytes
Latency-sensitive request percentage	65 % of total requests
Max permissible latency	50 - 700 ms
Latency threshold for latency-sensitive request	250 ms
Request generation rate	100 - 500 per second

Table 5.3 Different problem configurations

Parameter Name	Config 1	Config 2	Config 3	Config 4	Config 5
Number of vehicles	20	40	60	80	100
Requests per second	100 - 500	100 - 500	100 - 500	100 - 500	100 - 500

Table 5.4 Standard deviation of the obtained latency values

Parameter Name	This work	(Alvi et al., 2022)	(Yadav et al., 2020)	(Hameed et al., 2021)
Config 1	2.65430589	1.51995587	11.2232531	10.5909786
Config 2	1.3040544	3.31435265	16.9072412	25.7978003
Config 3	3.97123004	5.87752374	19.2917467	37.7000619
Config 4	4.26275534	9.25701405	69.4280101	61.9156538
Config 5	4.26275534	9.25701405	69.4280101	61.9156538

Table 5.5 Standard deviation of the obtained energy consumption values

Parameter Name	This work	(Alvi et al., 2022)	(Yadav et al., 2020)	(Hameed et al., 2021)
Config 1	6.119692934	9.578401473	19.53532151	12.92356262
Config 2	17.85449569	15.67213275	20.61775632	14.908081
Config 3	16.93895136	6.52370949	7.699065426	10.16688639
Config 4	29.43610576	21.68488736	22.33821391	13.59828248
Config 5	19.25530403	6.592201836	21.6399781	14.91832039

Table 5.6 Standard deviation of the obtained network usage values

Parameter Name	This work	(Alvi et al., 2022)	(Yadav et al., 2020)	(Hameed et al., 2021)
Config 1	32.4356406	19.3085085	22.1238309	23.4507258
Config 2	14.3379327	24.8237463	21.2567852	19.9205019
Config 3	19.9884787	25.7841984	25.9896675	30.8361319
Config 4	21.3399261	25.8708617	21.2780686	22.9147577
Config 5	24.774552	21.4576645	24.3309415	33.1313029

5.3 Result and discussion

To ascertain the suitability and practical applicability of the proposed technique for real-time latency-sensitive VFC applications, the results of simulation studies carried out on realistic vehicular data obtained via SUMO are presented and analyzed as follows. Each of the results depicted in the plots is an average of the 5 simulation runs for each simulation configuration.

5.3.1 Effect on Latency

The plot in Fig 5.5 depicts the latency values obtained for considered 5 configurations (an average of the five simulation runs for each configuration). For the proposed scheme the latency values obtained are within 10 percent from config 1 to config 5. The standard deviation values of the results obtained under each configuration for latency are presented in Table 5.4. On the contrary, for the ETE_{FN} scheme presented in (Alvi et al., 2022), the latency remains fairly close to the proposed scheme from config 1 to 4, and it increases sharply from config 4 to 5 (approximately 40 percent). For ECOS in (Yadav et al., 2020), the latency increases very sharply from config 3 to config 5 (approximately 237 percent), and for (Hameed et al., 2021), latency increases linearly from config 1 to config 4, and after that increases very sharply (approximately 302 percent). This research work assumes 65 percent of the total requests to be latency-sensitive

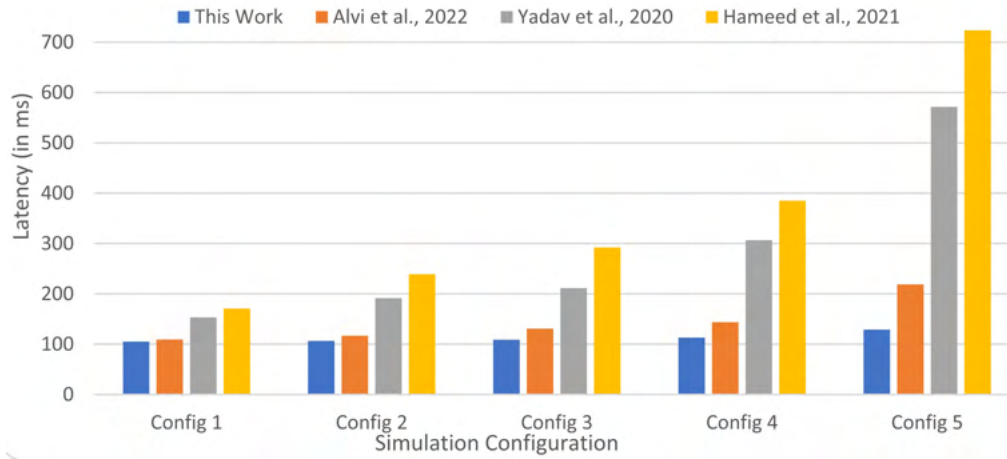


Figure 5.5 Latency values comparison of the proposed technique with previous competing works

and a 75 percent probability of processing at VFN. The latency graph of Fig 5.5 depicts that because of dynamic resource provisioning in the proposed technique (i.e., ACT-DW with a 75 percent probability of processing at VFN), all the latency-sensitive requests are being processed at VFN. Whereas, in the case of the other three schemes there is no dynamic resource provisioning. Thus, as the workload rises with number of vehicles rising from config 1 to config 5, a higher percentage of requests are offloaded to the cloud, resulting in the latency increasing very sharply due to the saturation of resources at VFN.

Fig 5.6 depicts the adaptive adjustment in the number of VFNs with the change in the number of VNs, and thus workload. As the workload increases, the number of VFNs in the proposed work also changes from 3 to 4, 5, and 6 for the number of VNs 20, 40, 60, 80, and 100, respectively, and vice-versa, based on ACT-DW. The novel concept of dynamic resource adaptation as per ACT-DW is responsible for the majority of latency-sensitive requests being processed at fog instead of offloading to the cloud.

5.3.2 Effect on Energy consumption

Total energy consumption is depicted in Fig 5.7 for different problem configurations. Energy consumed in proposed technique shows a linear increase with an increase in workload (i.e., from config 1 to 5). Analysis of the results il-

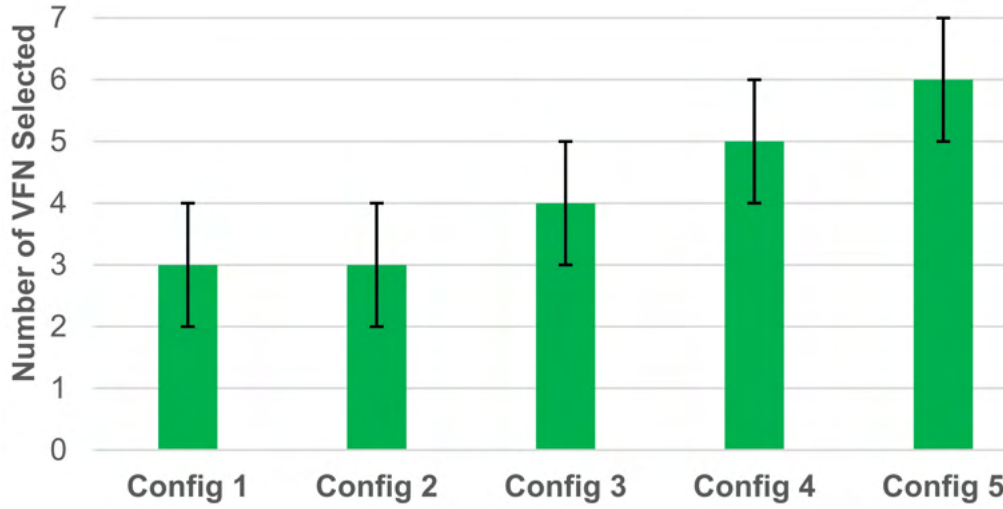


Figure 5.6 Variation in the number of VFNs selected by ACT-DW (dynamic resource adaptation based on the instantaneous workload demand)

illustrates that the proposed scheme has on average 7, 22, and, 28 percent less energy consumption as compared to (Alvi et al., 2022; Yadav et al., 2020), and (Hameed et al., 2021) respectively. The reduction in energy consumption in the proposed technique is because all the latency-sensitive requests are being processed at VFN, at all workloads because of the dynamic resource provisioning by ACT-DW. Total energy consumption consists of transmission energy and computational energy. VFN is connected at single-hop communication and has the least computational energy consumption among VFN, FCN, and cloud servers. However, in the case of the other three schemes, some requests are dispatched to cloud at varying workloads from config 1 to 5, resulting in higher energy consumption due to transmission and computational energy elements.

5.3.3 Effect on Network Bandwidth Consumption

Figure 5.8 depicts plot of the network bandwidth consumption for the proposed research work and the three competing works. Average network usage for the the proposed technique is 10, 19, and 48 percent lower as compared to (Alvi et al., 2022; Yadav et al., 2020), and (Hameed et al., 2021) respectively. In the proposed work, because of the novel concept of the pub-sub model (with very small size, event-triggered messages) being utilized for scheduling, minimum network



Figure 5.7 Energy consumption comparison

bandwidth is consumed for overhead message transmission during scheduling and management. Further, all the latency-sensitive requests (i.e., 65 percent of the total) are processed at VFN because of dynamic resource provisioning due to ACT-DW. This ensures that in proposed system network bandwidth consumption in the transmission of requests /replies from source to destination and vice-versa, are much lower than other three schemes wherein, a fraction of requests are processed in the cloud.

5.3.4 Effect on Maximum Latency Satisfaction

For real-time, latency-sensitive requests, it is most important that the maximum permissible latency limit is satisfied, otherwise, it may result in accidents or safety hazards on the road. To illustrate the practical applicability of this research work for real-time, latency-sensitive applications in VFC, the percentage of total requests satisfying the maximum permissible latency limits is studied under different problem configurations. Fig 5.9 depicts the percentage of requests satisfying the maximum latency limit for the proposed work is between 96 and 94, and for (Alvi et al., 2022) it is between 93 and 80. However, for (Yadav et al., 2020) and (Hameed et al., 2021) the percentage of requests satisfying maximum latency constraint drops rapidly, as the number of VNs increases more than 40 (i.e. config 3 to 5). In the proposed research work, all the latency-sensitive

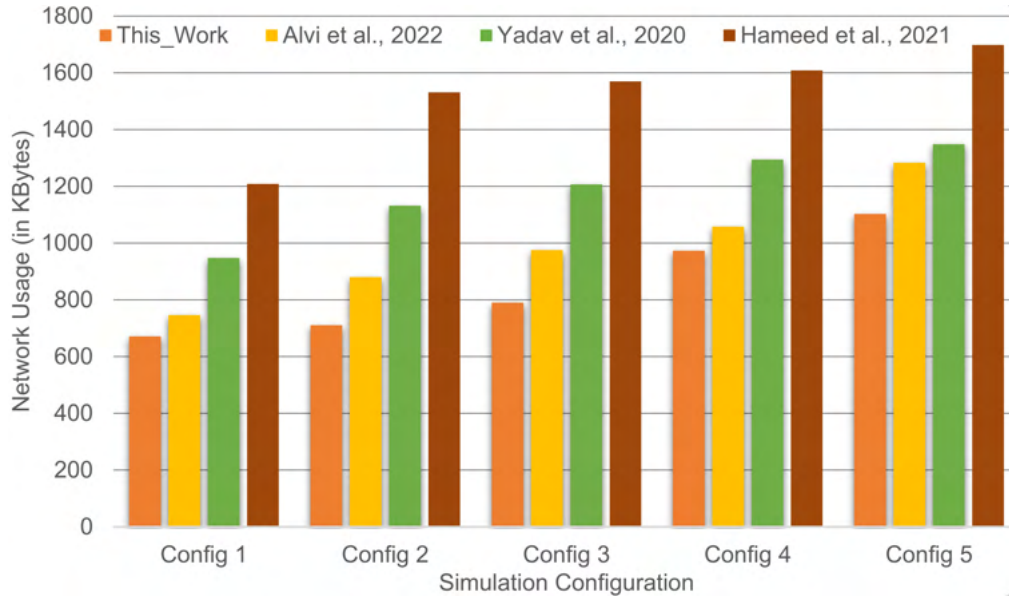


Figure 5.8 Network bandwidth consumption comparison

requests are processed at VFN (having the least latency among VFN, FCN, and cloud), because of the dynamic resource provisioning provided by ACT-DW. Whereas, in the other three schemes, a part of requests are dispatched to cloud for processing, with an increase in the workload. Hence, the QoS satisfaction ratio of the proposed technique is consistently high and the best among the compared schemes.

5.3.5 Statistical Validation of the Simulation Results

Levene's test is suitable for testing if the multiple data samples have equal variances. It is powerful, robust, and less sensitive to the sample's departure from normality for checking the homogeneity of variances in a group of three or more (O'Neill & Mathews, 2000; Vohra et al., 2011). In the proposed research work Levene's test has been conducted independently for each of the three parameters under study. The test has been conducted considering the simulation result for all the configurations (config 1-5) for each of the compared research works (i.e., four groups with 25 samples each) as a unique data set. Tables 5.7 to 5.12 present the summary and statistics of the Levene's test carried out for four competing pieces of research for three parameters namely latency, energy consumption, and network usage, respectively. The df (degree of freedom) between groups

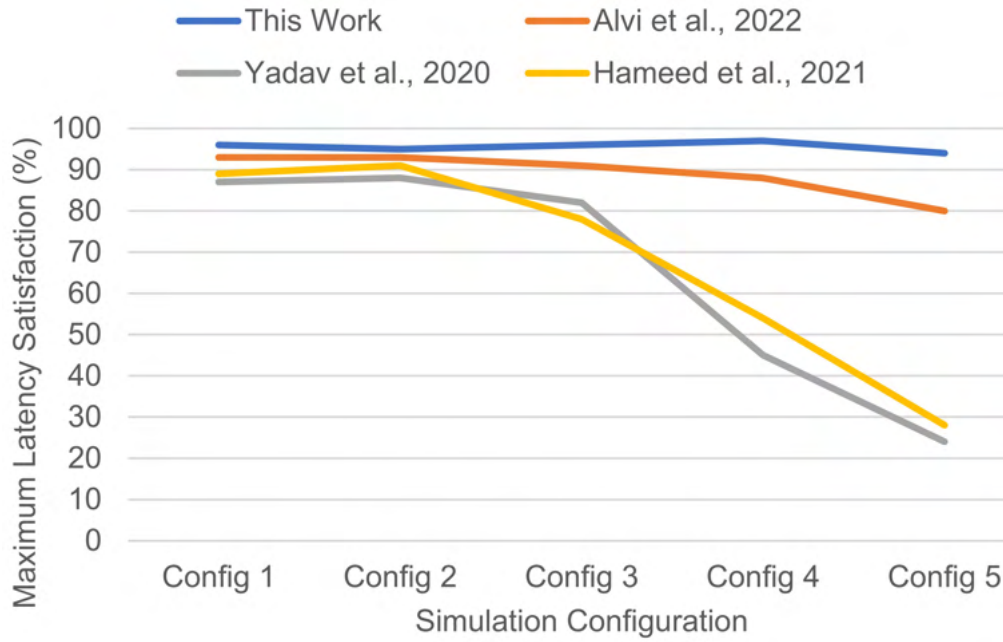


Figure 5.9 Maximum latency constraint satisfaction ratio

is obtained by calculating the number of groups minus one (i.e., $4-1=3$), and the degree of freedom within groups is total cases minus the number of groups (i.e., $100-4=96$). SS indicates squared sums and MS indicates mean squared sums of the samples. The computed p-value indicates the believability of the case that Null hypothesis is correct given the sample data. F (critical value) is obtained from F-table based on the values of DF1 and DF2 at the significance level ($\sigma = 0.05$) (Tiku, 1967). The decision on Levene's test results is taken as follows. If value of test statistic falls in rejection region i.e., $F(\text{observed}) \geq F(\text{critical})$ or if $p\text{-value} \leq \sigma$ then reject H_0 ; otherwise, fail to reject H_0 .

Table 5.7 Levene's test summary for latency results

Groups	Count	Sum	Average	Variance
This work	25	2817.524	112.7009	88.83651
(Alvi et al., 2022)	25	3600.005	144.0002	1639.406
(Yadav et al., 2020)	25	7168.593	286.7437	24967.09

(Hameed et al., 2021)	25	9991.53	399.6612	48252.9
-----------------------	----	---------	----------	---------

Table 5.8 Levene's test statistics for latency results ($\sigma = 0.05$)

Source of Variation	SS	df	MS	F (obs)	P-value	F crit
Between Groups	1325658.1	3	441886	23.58353	1.61E-11	2.699393
Within Groups	1798757.6	96	18737.06			
Total	3124415.7	99				

Table 5.9 Levene's test summary for energy consumption results

Groups	Count	Sum	Average	Variance
This work	25	4215.813	168.6325	1725.061
(Alvi et al., 2022)	25	4441.761	177.6704	1545.917
(Yadav et al., 2020)	25	5786.683	231.4673	813.8509
(Hameed et al., 2021)	25	5797.213	231.8885	1231.829

Table 5.10 Levene's test statistics for energy consumption results ($\sigma = 0.05$)

Source of Variation	SS	df	MS	F (obs)	P-value	F (crit)
---------------------	----	----	----	---------	---------	----------

Between Groups	86656.87	3	28885.62	21.73217	7.98E-11	2.699393
Within Groups	127599.8	96	1329.164			
Total	214256.6	99				

Table 5.11 Levene's test summary for network usage results

Groups	Count	Sum	Average	Variance
This work	25	21237.19	849.4877	28413.94
(Alvi et al., 2022)	25	24709.1	988.3639	34262.54
(Yadav et al., 2020)	25	29643.79	1185.752	21002.57
(Hameed et al., 2021)	25	36107.51	1306.623	19381.69

Table 5.12 Levene's test statistics for network usage results ($\sigma = 0.05$)

Source of Variation	SS	df	MS	F (obs)	P-value	F (crit)
Between Groups	6402995	3	2134332	75.31383	3.96E-25	2.699393
Within Groups	2720561	96	28339.17			
Total	9123556	99				

The null hypothesis is H_0 : the variances of all four groups are the same, and the alternate hypothesis is H_a : the variances of at least one group are significantly different than the other three groups. Confidence interval of 95 percent

is chosen empirically for all three sets of statistical validation simulations. In Table 5.8, for latency, the p-value obtained $1.61E-11$ is lower than σ , and F observed (23.58353) $>$ F critical (2.699393), indicating that there is enough evidence to reject the null hypothesis with 95% confidence. In Table 5.10, for energy consumption the p-value obtained $7.98E-11$ is lower than σ , and for network bandwidth consumption (21.73217) $>$ F critical (2.699393), indicating that there is enough evidence to reject the null hypothesis with 95 confidence. In Table 5.12, for network usage, the p-value obtained $3.96E-25$ is lower than σ , and F observed (75.31383) $>$ F critical (2.699393), indicating that there is enough evidence to reject the null hypothesis with 95% confidence. Therefore, for all the three parameters under study in this research work, Levene's test successfully validates the rejection of the homogeneity of variance. From the plots in Fig 5.5 - 5.9 and the statistical validation test it is confirmed that the proposed technique is quantitatively better than the other three researches under all simulation configurations (config 1-5) for latency, energy consumption, and network usage parameters.

5.4 Summary

VFC is an important research area that promises a solution to many of the open research problems in ITS. One prominent problem being researched is reducing latency while optimizing resource utilization in VFC. In this chapter, the proposed technique for optimal vehicular fog node selection for processing latency-sensitive application request is validated using simulation. Proposed technique is compared with competing research works. Analysis of results proves that present technique performs better, on latency, resource consumption, and overall QoS implementation for latency-sensitive requests. The statistical validation test of results also reaffirm the quantitative advantages of the proposed technique.

Chapter 6

Conclusion and Future Research Suggestions

6.1 Introduction

This chapter of the thesis summarizes the complete research work, and presents the future research scope related to this research work. The Section 6.2 provides summary of the research findings. Section 6.3 highlights important contributions of this research work. Section 6.4 presents the limitations of this research work and provides the directions for furthering the research in the same domain based on the contributions of this work.

In past decade, unprecedented rate of development in the fields of ICT led to development of smart, connected vehicles. This further led to the evolution of the concept of autonomous driving and the need for ubiquitous computing services, including those in moving vehicles. Accordingly, the demand for vehicular services and applications is growing very fast. Vehicular applications are primarily of two types namely, latency-sensitive (e.g., safety-critical applications) and latency-tolerant (e.g., all other applications like entertainment, advertising, route planning, etc.). With the growing demand for safety-critical applications, the issue of meeting latency and QoS requirements for latency-sensitive applications becomes very pertinent to ensure safety on the road. In last few years, VFC evolved as crucial research area in addressing this critical issue. Motivated by this important socio-technological challenge of future smart cities, this

doctoral research work has contributed significantly to addressing the issue of optimizing latency and energy for latency-sensitive vehicular applications in ITS.

6.2 Summary of Research Findings

Summary of research finding of this doctoral research work is as follows:-

In this research work, we started with survey of existing research work in this domain. From the analysis of the published research work important characteristics of the optimal vehicular fog node for processing a latency-sensitive request were identified. primary goal of this research work is to choose the optimal VFN for processing latency-sensitive requests. Accordingly, a MOO model for the three-tier VFC architecture was developed. The validation of the MOO model was conducted using the MATLAB toolbox, 'fmincon' by taking latency as a decision parameter. We have used a one-way ANOVA test for statistical validation of the simulation data in this simulation setup.

Based on the MOO model developed, the algorithms (i.e., LEAQoS, and ACT-DW) for optimal VFN selection were developed. The LEAQoS algorithm helps select the most optimum fog node available for processing the current latency-sensitive request. The ACT-DW algorithm adaptively adjusts the total available computational resource pool at VFC, as per the instantaneous demand of computational resources. Hence both these proposed algorithms facilitate chose optimal fog nodes in VFC to reduce energy consumed and maximum permissible latency constraint. The simulation setup uses realistic vehicular traffic data (obtained through the SUMO tool) from Chandigarh City, India, and synthetically generated application request data for validation of the proposed technique. Simulation confirms efficacy of presented technique. The comparative study proves that the proposed technique outperforms competing works concerning latency and energy consumption. The maximum latency satisfaction ratio for the proposed technique is between 94-96 percent as compared to 80-93 percent for the nearest competing work. Levene's F-test (one-tailed) with a 95 percent confidence interval is executed for validation of results.

6.3 Contributions of the Research Work

Important outcomes of this doctoral research work are as follows:-

- Research investigation is conducted on characteristics of optimal Vehicular Fog Nodes for identifying the mandatory constrained parameters.
- A mathematical model is designed to represent these characteristics in a multivariate constrained non-linear multi-objective optimization (MOO) problem.
- The multi-variable, multi-constraint model is solved using the Interior-point optimization algorithm for achieving the goal of optimum latency under mutually conflicting constraints of controlled parameters like radio bandwidth, coverage range of a node, number of vehicles, and discreet workload sizes.
- Evaluation and comparison of the latency and QoS satisfaction ratio of the generic fog-cloud model, *ECOS*, and *ETE_{FN}* techniques to the proposed VFC model for different workload sizes, vehicle fleet sizes, and request arrival rate of VFN by applying thresholds to constrained variables, for demonstrating quantitative QoS enhancement of VFC model over fog-cloud and other state-of-the-art schemes.
- A novel technique (i.e., heuristic-based LEAQoS algorithm) is proposed for resource optimization in VFC while satisfying the maximum permissible latency parameter for real-time, latency-sensitive requests.
- An adaptive capacity tuning algorithm for the dynamic workload (ACTDW) is presented, to adaptively adjust the real-time availability of VFC resources, based on the instantaneous workload, thus, minimizing the latency, and cost to the VFC service provider.
- A unique, nature-inspired pub-sub model is proposed for computing of-flooding, with algorithm processing at FCN, which ensures minimum com-

pute overhead at VFN. Thus, minimizing network bandwidth consumption, and maximum processing power of VFNs is available for processing latency-sensitive requests, ensuring maximum QoS satisfaction.

- A comparative analysis of the proposed LEAQoS technique with peer-competing techniques proves the suitability of the proposed technique for real-time, latency-sensitive requests in VFC.

6.4 Limitations and Future Research

There are a few assumptions made during the simulation in this research as brought out in relevant chapters. Most important of are assumption of seamless connectivity between the nodes in a radio communication scenario. Accordingly, to limit the scope of the study and focus on the relevant issue under discussed in this research work, the packet drop in wireless communication scenario had not been studied as part of this research work.

6.4.1 Limitations of the Sample Data used during Simulation

The vehicular traffic data for the simulation has been obtained using the SUMO tool from the OSM map, and the vehicular application data is synthetically generated. The results presented are from extensive simulation runs under different test conditions defined in the research work. However, to ensure the practical applicability of the proposed technique it would be essential to run the practical experiments with real-world vehicular traffic and application data.

6.4.2 Evolution of Research

Firstly, the presented technique for a dynamic vehicular fog node selection from the pool of all vehicular nodes in a VFC scenario has been proven to be quantitatively better as compared to the previously presented research concerning latency and energy consumption based on the simulation results. However, practical experimentation with real-world data would be necessary before putting the presented

technique into action in connected cars. Secondly, exponential rise in number of connected cars, the criticality and complexity of the optimal VFN selection for processing latency-sensitive, safety-critical application requests are increasing. With the proven benefits of ML techniques for complex problem solving, it is proposed that the use of ML algorithms be explored in future research works for achieving the most optimum solutions for VFN selection in vehicular networks.

References

Alam, M., Ferreira, J., & Fonseca, J. (2016). Introduction to intelligent transportation systems. *Studies in Systems, Decision and Control*, 52, 1–17. doi: 10.1007/978-3-319-28183-4_1

Alamer, A., & Basudan, S. (2020). An efficient truthfulness privacy-preserving tendering framework for vehicular fog computing. *Engineering Applications of Artificial Intelligence*. doi: 10.1016/j.engappai.2020.103583

Alharbi, H. A., Elgorashi, T. E., & Elmirghani, J. M. (2020). Energy efficient virtual machines placement over cloud-fog network architecture. *IEEE Access*, 8, 94697–94718. doi: 10.1109/ACCESS.2020.2995393

Ali, Oy, N., Dudek, M., Jackowiak, P., Poplawski, A., & Oy, N. (2020). (12) United States Patent. , 2.

Alouini, M.-S., & Goldsmith, A. (1997). Capacity of nakagami multipath fading channels. In *1997 IEEE 47th vehicular technology conference. technology in motion* (Vol. 1, pp. 358–362).

Alvi, A. N., Javed, M. A., Hasanat, M. H. A., Khan, M. B., Saudagar, A. K. J., Alkathami, M., & Farooq, U. (2022). Intelligent Task Offloading in Fog Computing Based Vehicular Networks. *Applied Sciences (Switzerland)*, 12(9). doi: 10.3390/app12094521

Angus, I. (1946). An Introduction to Erlang B and Erlang C. *Telemanagement*(187), 6–9. Retrieved from <http://voip.poly.ro/docs/ErlangB&C.pdf>

Anjum, A., & Khan, A. A. (2019). Optimized Communication in 5G-Driven Vehicular Ad-hoc Networks (VANETs).

- Baccarelli, E., Naranjo, P. G., Scarpiniti, M., Shojafar, M., & Abawajy, J. H. (2017). Fog of Everything: Energy-Efficient Networked Computing Architectures, Research Challenges, and a Case Study. *IEEE Access*, 5, 9882–9910. doi: 10.1109/ACCESS.2017.2702013
- Bai, F., & Krishnan, H. (2006). Reliability analysis of DSRC wireless communication for vehicle safety applications. In *Ieee conference on intelligent transportation systems, proceedings, itsc*. doi: 10.1109/itsc.2006.1706767
- Baltrunas, D., Elmokashfi, A., Kvalbein, A., & Alay, Ö. (2016). Investigating packet loss in mobile broadband networks under mobility. *2016 IFIP Networking Conference (IFIP Networking) and Workshops, IFIP Networking 2016*, 225–233. doi: 10.1109/IFIPNetworking.2016.7497225
- Barton, M. I. L. F. R., Goren, M. J. M. N., & Mahmoudi, C. T. (2018). Fog Computing Conceptual Model: Recommendations of the National Institute of Standards and Technology. *NIST Special Publication 500-325, 1(2)*. doi: <https://doi.org/10.6028/NIST.SP.500-325>
- Behrisch, M., Bieker, L., Simulation, J. E. . . . i. S., & undefined 2011. (n.d.). SUMO—simulation of urban mobility: an overview. *elib.dlr.de*. Retrieved from <https://elib.dlr.de/71460/>
- Bennis, M., Debbah, M., & Poor, H. V. (2018). Ultrareliable and Low-Latency Wireless Communication: Tail, Risk, and Scale. *Proceedings of the IEEE*, 106(10), 1834–1853. doi: 10.1109/JPROC.2018.2867029
- Bennis, M., Debbah, M., Poor, H. V., Baltrunas, D., Elmokashfi, A., Kvalbein, A., . . . Chen, C. (2016). Data-driven intelligent transportation systems: A survey. *IEEE Transactions on Intelligent Transportation Systems*, 12(4), 1624–1639. doi: 10.1109/TITS.2011.2158001
- Binwal, D. C., & Kapoor, M. (2022). A Survey on Architecture, Applications, and Challenges in Vehicular Fog Computing. , *International Journal of Sensors, Wireless Communications and Control*, 12. Retrieved from <https://dx.doi.org/10.2174/2210327912666220127130014> doi: 10.2174/2210327912666220127130014

Birhanie, H. M., Senouc, S. M., Messous, M. A., Arfaoui, A., & Kies, A. (2020). A Stochastic Theoretical Game Approach for Resource Allocation in Vehicular Fog Computing. *2020 IEEE 17th Annual Consumer Communications and Networking Conference, CCNC 2020*, 18–19. doi: 10.1109/CCNC46108.2020.9045224

Bonomi, F., Milito, R., Zhu, J., & Addepalli, S. (2012). Fog computing and its role in the internet of things. *MCC'12 - Proceedings of the 1st ACM Mobile Cloud Computing Workshop*, 13–15. doi: 10.1145/2342509.2342513

Boualouache, A., Technology, H. S. . . . o. V., undefined 2021, Sedjelmaci, H., Engel, T., Technology, H. S. . . . o. V., & undefined 2021. (2021). Consortium Blockchain for Cooperative Location Privacy Preservation in 5G-Enabled Vehicular Fog Computing. *IEEE Transactions on Vehicular Technology*, 70(7), 7087–7102. Retrieved from <https://ieeexplore.ieee.org/abstract/document/9440815/> doi: 10.1109/TVT.2021.3083477

Brooks, G. P., & Johanson, G. A. (2011). Sample size considerations for multiple comparison procedures in ANOVA. *Journal of Modern Applied Statistical Methods*, 10(1), 97–100. doi: 10.22237/jmasm/1304222940

Byers, C. C. (2017). Architectural Imperatives for Fog Computing: Use Cases, Requirements, and Architectural Techniques for Fog-Enabled IoT Networks. *IEEE Communications Magazine*, 55(8), 14–20. doi: 10.1109/MCOM.2017.1600885

Calheiros, R. N., Ranjan, R., Beloglazov, A., De Rose, C. A., & Buyya, R. (2011, jan). CloudSim: A toolkit for modeling and simulation of cloud computing environments and evaluation of resource provisioning algorithms. *Software - Practice and Experience*, 41(1), 23–50. doi: 10.1002/SPE.995

Chen, C., Qiu, T., Hu, J., Ren, Z., Zhou, Y., & Sangaiah, A. K. (2017). A congestion avoidance game for information exchange on intersections in heterogeneous vehicular networks. *Journal of Network and Computer Applications*, 85, 116–126. Retrieved from <http://dx.doi.org/10.1016/j.jnca.2016.12.014> doi: 10.1016/j.jnca.2016.12.014

Cheng, J., Cheng, J., Zhou, M., Liu, F., Gao, S., & Liu, C. (2015). Routing in internet of vehicles: A review. *IEEE Transactions on Intelligent Transportation Systems*. doi: 10.1109/TITS.2015.2423667

Cheng, X., Chen, C., Zhang, W., & Yang, Y. (2017). 5G-Enabled Cooperative Intelligent Vehicular When Benz Meets Marconi. *IEEE Computer Society*(August), 53–59.

Coleman, T., Branch, M. A., & Grace, A. (2009). Optimization Toolbox Matlab 7.8.0.

Consortium, O., & Working, A. (2017). Open fog reference architecture for fog computing. *Open Fog Consortium Architecture Working Group*(February), 1–162.

Contreras-Castillo, J., Zeadally, S., & Guerrero-Ibanez, J. A. (2018). Internet of Vehicles: Architecture, Protocols, and Security. *IEEE Internet of Things Journal*, 5(5), 3701–3709. doi: 10.1109/JIOT.2017.2690902

Darwish, T. S., & Abu Bakar, K. (2018). Fog Based Intelligent Transportation Big Data Analytics in The Internet of Vehicles Environment: Motivations, Architecture, Challenges, and Critical Issues. *IEEE Access*, 6, 15679–15701. doi: 10.1109/ACCESS.2018.2815989

Datta, S. K., Da Costa, R. P. F., Harri, J., & Bonnet, C. (2016). Integrating connected vehicles in Internet of Things ecosystems: Challenges and solutions. *WoWMoM 2016 - 17th International Symposium on a World of Wireless, Mobile and Multimedia Networks*. doi: 10.1109/WoWMoM.2016.7523574

Din, S., Paul, A., & Rehman, A. (2019). 5G-enabled Hierarchical architecture for software-defined intelligent transportation system. *Computer Networks*, 150, 81–89. Retrieved from <https://doi.org/10.1016/j.comnet.2018.11.035> doi: 10.1016/j.comnet.2018.11.035

Ejaz, M., Kumar, T., Ylianttila, M., & Harjula, E. (2020). Performance and Efficiency Optimization of Multi-layer IoT Edge Architecture. (March).

Eltoweissy, M., Olariu, S., & Younis, M. (2010). Towards autonomous vehicular clouds: A position paper (Invited paper). *Lecture Notes of the Institute for*

Computer Sciences, Social-Informatics and Telecommunications Engineering, 49 LNICST, 1–16. doi: 10.1007/978-3-642-17994-5_1

Fan, W., Shi, Y., Chen, S., & Zou, L. (2012). A mobility metrics based dynamic clustering algorithm for vanets. *IET Conference Publications*, 2011(586 CP), 752–756. doi: 10.1049/cp.2011.0769

Flavio Bonomi, Rodolfo Milito, P. N., & Zhu, J. (2014). Fog Computing: A Platform for Internet of Things and Analytics. *Studies in Computational Intelligence*, 546, 1–19. doi: 10.1007/978-3-319-05029-4

Fu, F., Kang, Y., Zhang, Z., Yu, F. R., & Wu, T. (2020). Transcoding for live streaming-based on vehicular fog computing: An actor-critic DRL approach. In *Ieee infocom 2020 - ieee conference on computer communications workshops, infocom wkshps 2020*. doi: 10.1109/jiot.2020.3003398

Gasmi, R., & Aliouat, M. (2019). Vehicular Ad Hoc NETWORKS versus Internet of Vehicles-A Comparative View. *Proceedings - ICNAS 2019: 4th International Conference on Networking and Advanced Systems*, 0–5. doi: 10.1109/ICNAS.2019.8807870

Ge, S., Cheng, M., & Zhou, X. (2020). Interference Aware Service Migration in Vehicular Fog Computing. *IEEE Access*. doi: 10.1109/ACCESS.2020.2992275

Ghosh, K., Das, P. K., & Neogy, S. (2016). Kps: A fermat point based energy efficient data aggregating routing protocol for multi-sink wireless sensor networks. In R. Chaki, A. Cortesi, K. Saeed, & N. Chaki (Eds.), *Advanced computing and systems for security: Volume 1* (pp. 203–221). New Delhi: Springer India. Retrieved from https://doi.org/10.1007/978-81-322-2650-5_13 doi: 10.1007/978-81-322-2650-5_13

Ghosh, K., Neogy, S., Das, P. K., & Mehta, M. (2017). On regulating lifetime of a 3-sink wireless sensor network deployed for precision agriculture. *International Journal of Next-Generation Computing*, 8(3), 153–170.

Gross, M. (2016). A planet with two billion cars. *Current Biology*, 26(8), R307–R310. Retrieved from <http://dx.doi.org/10.1016/j.cub.2016.04.019> doi: 10.1016/j.cub.2016.04.019

Grover Jyoti, Jain Ashish, B. S. (2018). Real-Time VANET Applications Using Fog Computing; Proceedings of First International Conference on Smart System, Innovations and Computing. , 79(January). Retrieved from <http://link.springer.com/10.1007/978-981-10-5828-8> doi: 10.1007/978-981-10-5828-8

Gupta, H., Vahid Dastjerdi, A., Ghosh, S. K., & Buyya, R. (2017). iFogSim: A toolkit for modeling and simulation of resource management techniques in the Internet of Things, Edge and Fog computing environments. *Software - Practice and Experience*, 47(9), 1275–1296. doi: 10.1002/spe.2509

Haj Qasem, M., Abu-Srhan, A., Natoureh, H., & Alzaghouli, E. (2020). Fog Computing Framework for Smart City Design. *International Journal of Interactive Mobile Technologies (IJIM)*, 14(01), 109. doi: 10.3991/ijim.v14i01.9762

Hameed, A. R., ul Islam, S., Ahmad, I., & Munir, K. (2021). Energy- and performance-aware load-balancing in vehicular fog computing. *Sustainable Computing: Informatics and Systems*, 30, 100454. Retrieved from <https://doi.org/10.1016/j.suscom.2020.100454> doi: 10.1016/j.suscom.2020.100454

Hong, H. J., Tsai, P. H., & Hsu, C. H. (2016). Dynamic module deployment in a fog computing platform. *18th Asia-Pacific Network Operations and Management Symposium, APNOMS 2016: Management of Softwarized Infrastructure - Proceedings*, 1–6. doi: 10.1109/APNOMS.2016.7737202

Hoque, M. A., & Hasan, R. (2019). Towards an Analysis of the Architecture, Security, and Privacy Issues in Vehicular Fog Computing. *Conference Proceedings - IEEE SOUTHEASTCON, 2019-April*. doi: 10.1109/SoutheastCon42311.2019.9020476

Hou, X., Li, Y., Chen, M., Wu, D., Jin, D., & Chen, S. (2016). Vehicular Fog Computing: A Viewpoint of Vehicles as the Infrastructures. *IEEE Transactions on Vehicular Technology*, 65(6), 3860–3873. doi: 10.1109/TVT.2016.2532863

Hu, H. (2008). Poisson distribution and application. *University of Tennessee, Knoxville*, 0(5), 1–3.

Hu, P., Dhelim, S., Ning, H., & Qiu, T. (2017). Survey on fog computing: architecture, key technologies, applications and open issues. *Journal of Network and Computer Applications*, 98(April), 27–42. Retrieved from <http://dx.doi.org/10.1016/j.jnca.2017.09.002> doi: 10.1016/j.jnca.2017.09.002

Huang, C., Lu, R., & Choo, K. K. R. (2017). Vehicular Fog Computing: Architecture, Use Case, and Security and Forensic Challenges. *IEEE Communications Magazine*, 55(11), 105–111. doi: 10.1109/MCOM.2017.1700322

Huang, J., Qian, Y., & Hu, R. Q. (2020). Security Provision for Vehicular Fog Computing. *IEEE Vehicular Technology Conference, 2020-May*, 16–20. doi: 10.1109/VTC2020-Spring48590.2020.9129424

Hussain, M., Saad Alam, M., Sufyan Beg, M. M., & Akhtar, N. (2020). Towards minimizing delay and energy consumption in vehicular fog computing (VFC). *Journal of Intelligent Fuzzy Systems*, 38(5), 6549–6560. doi: 10.3233/jifs-179735

Hussain, M. M., & Beg, M. M. (2021). CODE-V: Multi-hop computation offloading in Vehicular Fog Computing. *Future Generation Computer Systems*, 116, 86–102. Retrieved from <https://doi.org/10.1016/j.future.2020.09.039> doi: 10.1016/j.future.2020.09.039

Hussain, R., Son, J., Eun, H., Kim, S., & Oh, H. (2012). Rethinking Vehicular Communications: Merging VANET with cloud computing. *CloudCom 2012 - Proceedings: 2012 4th IEEE International Conference on Cloud Computing Technology and Science*, 606–609. doi: 10.1109/CloudCom.2012.6427481

Hussain, R., & Zeadally, S. (2019). Autonomous Cars: Research Results, Issues, and Future Challenges. *IEEE Communications Surveys and Tutorials*, 21(2), 1275–1313. doi: 10.1109/COMST.2018.2869360

Ieee, S. M., Ieee, F., Zhang, J., & Letaief, K. B. (2019). Mobile edge intelligence and computing for the internet of vehicles. *arXiv*(March). doi: 10.1109/JPROC.2019.2947490

- Iqbal, S., Malik, A. W., Rahman, A. U., & Noor, R. M. (2020). Blockchain-based reputation management for task offloading in micro-level vehicular fog network. *IEEE Access*. doi: 10.1109/ACCESS.2020.2979248
- Kai, K., Cong, W., & Tao, L. (2016). Fog computing for vehicular Ad-hoc networks: Paradigms, scenarios, and issues. *Journal of China Universities of Posts and Telecommunications*, 23(2), 56–65,96. Retrieved from [http://dx.doi.org/10.1016/S1005-8885\(16\)60021-3](http://dx.doi.org/10.1016/S1005-8885(16)60021-3) doi: 10.1016/S1005-8885(16)60021-3
- Kaiwartya, O., Abdullah, A. H., Cao, Y., Altameem, A., Prasad, M., Lin, C. T., & Liu, X. (2016). Internet of Vehicles: Motivation, Layered Architecture, Network Model, Challenges, and Future Aspects. *IEEE Access*. doi: 10.1109/ACCESS.2016.2603219
- Keselman, H. J., Huberty, C. J., Cribbie, R. A., Lowman, L. L., Lix, L. M., Donahue, B., . . . Levin, J. R. (1998). Statistical practices of educational researchers: An analysis of their ANOVA, MANOVA, and ANCOVA analyses. *Review of Educational Research*, 68(3), 350–386. doi: 10.3102/00346543068003350
- Khaliq, K. A., Chughtai, O., Shahwani, A., Qayyum, A., & Pannek, J. (2019). Road accidents detection, data collection and data analysis using V2X communication and edge/cloud computing. *Electronics (Switzerland)*, 8(8). doi: 10.3390/electronics8080896
- Khan, A. A., Abolhasan, M., & Ni, W. (2018). 5G next generation VANETs using SDN and fog computing framework. *CCNC 2018 - 2018 15th IEEE Annual Consumer Communications and Networking Conference, 2018-Janua*, 1–6. doi: 10.1109/CCNC.2018.8319192
- Khattak, H. A., Farman, H., Jan, B., Ud Din, I., Hasan Ali Khattak, Haleem Farman, Bilal Jan, & Din, I. U. (2019). Toward Integrating Vehicular Clouds with IoT for Smart City Services. *IEEE Network*, 33(2), 65–71. Retrieved from <https://doi.org/10.1016/j.future.2018.07.040><http://arxiv.org/abs/1908.06849><https://doi.org/10.1016/j.adhoc.2020.102150> doi: 10.1109/MNET.2019.1800236

- Kong, Q., Su, L., & Ma, M. (2020). Achieving Privacy-Preserving and Verifiable Data Sharing in Vehicular Fog With Blockchain. *IEEE Transactions on Intelligent Transportation Systems*. doi: 10.1109/tits.2020.2983466
- Kui, X., Sun, Y., Zhang, S., & Li, Y. (2018). Characterizing the Capability of Vehicular Fog Computing in Large-scale Urban Environment. *Mobile Networks and Applications*, 23(4), 1050–1067. doi: 10.1007/s11036-017-0969-8
- Li, H., Ota, K., & Dong, M. (2019). Deep reinforcement scheduling for mobile crowdsensing in fog computing. *ACM Transactions on Internet Technology*, 19(2). doi: 10.1145/3234463
- Li, M., Yang, S., & Liu, X. (2016). Pareto or non-pareto: Bi-criterion evolution in multiobjective optimization. *IEEE Transactions on Evolutionary Computation*, 20(5), 645-665. doi: 10.1109/TEVC.2015.2504730
- Li, M., Zhu, L., Zhang, Z., Du, X., & Guizani, M. (2018). PROS: A privacy-preserving route-sharing service via vehicular fog computing. *IEEE Access*. doi: 10.1109/ACCESS.2018.2878792
- Liao, H., Mu, Y., Zhou, Z., Sun, M., Wang, Z., & Pan, C. (2020). Blockchain and Learning-Based Secure and Intelligent Task Offloading for Vehicular Fog Computing. *IEEE Transactions on Intelligent Transportation Systems*. doi: 10.1109/tits.2020.3007770
- Liao, H., Zhou, Z., Zhao, X., Ai, B., & Mumtaz, S. (2019). Task offloading for vehicular fog computing under information uncertainty: A matching-learning approach. In *2019 15th international wireless communications and mobile computing conference, iwcmc 2019*. doi: 10.1109/IWCMC.2019.8766579
- Liu, B., Wang, Z., Qin, J., Jiang, Y., Chen, X., Wang, E., & Xiong, S. (2021). A novel framework for message dissemination with consideration of destination prediction in VFC. *Neural Computing and Applications*, 9. Retrieved from <https://doi.org/10.1007/s00521-021-05754-9> doi: 10.1007/s00521-021-05754-9
- Liu, L., Chen, C., Pei, Q., Maharjan, S., & Zhang, Y. (2019). Vehicular Edge Computing and Networking: A Survey. , XX(Xx), 1–19. Retrieved from <http://arxiv.org/abs/1908.06849>

- Liu, S., Xiang, W., & Punithan, M. X. (2018). An Empirical Study on Performance of DSRC and LTE-4G for Vehicular Communications. *IEEE Vehicular Technology Conference, 2018-Augus*, 1–5. doi: 10.1109/VTCFall.2018.8690979
- Mahmud, R., Pallewatta, S., Goudarzi, M., & Buyya, R. (2021). IFogSim2: An Extended iFogSim Simulator for Mobility, Clustering, and Microservice Management in Edge and Fog Computing Environments. Retrieved from <http://arxiv.org/abs/2109.05636>
- Mahmud, R., Ramamohanarao, K., & Buyya, R. (2018). Latency-aware application module management for fog computing environments. *ACM Transactions on Internet Technology*, 19(1). doi: 10.1145/3186592
- Mahmud, R., Ramamohanarao, K., & Buyya, R. (2020). Application Management in Fog Computing Environments: A Taxonomy, Review and Future Directions. *ACM Computing Surveys*, 53(4). doi: 10.1145/3403955
- Marín-Tordera, E., Masip-Bruin, X., García-Almiñana, J., Jukan, A., Ren, G. J., & Zhu, J. (2017). Do we all really know what a fog node is? Current trends towards an open definition. *Computer Communications*, 109, 117–130. doi: 10.1016/j.comcom.2017.05.013
- Mattia, G. P., Pietrabissa, A., & Beraldi, R. (2023). A Load Balancing Algorithm for Equalising Latency across Fog or Edge Computing Nodes. , 14(8). doi: 10.1109/TSC.2023.3265883
- Mayer, R., Graser, L., Gupta, H., Saurez, E., & Ramachandran, U. (2018). EmuFog: Extensible and scalable emulation of large-scale fog computing infrastructures. *2017 IEEE Fog World Congress, FWC 2017*, 1–6. doi: 10.1109/FWC.2017.8368525
- Mekki, T., Jabri, I., Rachedi, A., & Ben Jemaa, M. (2018). Towards Multi-Access Edge Based Vehicular Fog Computing Architecture. *2018 IEEE Global Communications Conference, GLOBECOM 2018 - Proceedings*, 1–6. doi: 10.1109/GLOCOM.2018.8647850

- Mekki, T., Jmal, R., Chaari, L., Jabri, I., & Rachedi, A. (2020). Vehicular Fog Resource Allocation Scheme: A Multi-Objective Optimization based Approach. *2020 IEEE 17th Annual Consumer Communications and Networking Conference, CCNC 2020*. doi: 10.1109/CCNC46108.2020.9045361
- Mendiboure, L., Chalouf, M. A., & Krief, F. (2019). Edge Computing Based Applications in Vehicular Environments: Comparative Study and Main Issues. *Journal of Computer Science and Technology*, 34(4), 869–886. doi: 10.1007/s11390-019-1947-3
- Meneguet, R. I., Boukerche, A., & Pimenta, A. H. (2019). AVARAC: An Availability-Based Resource Allocation Scheme for Vehicular Cloud. *IEEE Transactions on Intelligent Transportation Systems*, 20(10), 3688–3699. doi: 10.1109/TITS.2018.2880298
- Menon, V. G., & Prathap, J. (2017). Vehicular Fog Computing. *International Journal of Vehicular Telematics and Infotainment Systems*, 1(2), 15–23. doi: 10.4018/ijvtis.2017070102
- Mohanty, P., Kumar, L., Malakar, M., Vishwakarma, S. K., & Reza, M. (2018). Dynamic resource allocation in Vehicular cloud computing systems using game theoretic based algorithm. *PDGC 2018 - 2018 5th International Conference on Parallel, Distributed and Grid Computing(Vc)*, 476–481. doi: 10.1109/PDGC.2018.8745913
- Ning, Z., Huang, J., & Wang, X. (2019). Vehicular fog computing: Enabling real-time traffic management for smart cities. *IEEE Wireless Communications*, 26(1), 87–93. doi: 10.1109/MWC.2019.1700441
- O'Neill, M. E., & Mathews, K. (2000). A weighted least squares approach to Levene's test of homogeneity of variance. *Australian and New Zealand Journal of Statistics*, 42(1), 81–100. doi: 10.1111/1467-842X.00109
- OpenFog Consortium Architecture Working Group. (2017). OpenFog Reference Architecture for Fog Computing. *OpenFog*(February), 1–162. Retrieved from https://www.iiconsortium.org/pdf/OpenFogReferenceArchitecture_2_09_17.pdf doi: OPFRA001.020817

- Peng, X., Ota, K., & Dong, M. (2020). Multiattribute-Based Double Auction Toward Resource Allocation in Vehicular Fog Computing. *IEEE Internet of Things Journal*. doi: 10.1109/JIOT.2020.2965009
- Rahman, F. H., Newaz, S. H., Au, T. W., Suhaili, W. S., Mahmud, M. A., & Lee, G. M. (2022). EnTruVe: ENergy and TRUst-aware Virtual Machine allocation in VEHICLE fog computing for catering applications in 5G. *Future Generation Computer Systems*, 126, 196–210. Retrieved from <https://doi.org/10.1016/j.future.2021.07.036> doi: 10.1016/j.future.2021.07.036
- Rahman, F. H., Yura Muhammad Iqbal, A., Newaz, S. H., Thien Wan, A., & Ahsan, M. S. (2019). Street Parked Vehicles Based Vehicular Fog Computing: TCP Throughput Evaluation and Future Research Direction. *International Conference on Advanced Communication Technology, ICACT, 2019-Febru*, 26–31. doi: 10.23919/ICACT.2019.8701912
- Ran, M., & Bai, X. (2020). Vehicle cooperative network model based on hypergraph in vehicular fog computing. *Sensors (Switzerland)*, 20(8). doi: 10.3390/s20082269
- Sadiku, M. N. O., Tembely, M., & Musa, S. M. (2018). Internet of Vehicles: an Introduction. *International Journal of Advanced Research in Computer Science and Software Engineering*, 8(1), 11. doi: 10.23956/ijarcsse.v8i1.512
- Sami, H., & Mourad, A. (2020). Dynamic On-Demand Fog Formation Offering On-the-Fly IoT Service Deployment. *IEEE Transactions on Network and Service Management*, 17(2), 1026–1039. doi: 10.1109/TNSM.2019.2963643
- Sathish, K., & Kumar, T. A. (2020). Resource Management Techniques in Vehicular Fog Computing : A Brief Survey. , 4(June), 26–34.
- Satyanarayanan, M. (2017). The Emergence of Edge Computing. *COMPUTER PUBLISHED BY THE IEEE COMPUTER SOCIETY*(0018), 17–20.
- Sayed, M. M., & Kashkoush, M. S. (2020). Towards Resilient Adaptive Vehicular Fog Computing. , 681–685.
- Shaheen, Q., Shiraz, M., Hashmi, M. U., Mahmood, D., Zhiyu, Z., & Akhtar, R. (2020). A Lightweight Location-Aware Fog Framework (LAFF) for QoS

in Internet of Things Paradigm. *Mobile Information Systems*, 2020. doi: 10.1155/2020/8871976

Shi, M., Lu, C., Zhang, Y., & Yao, D. (2017). DSRC and LTE-V communication performance evaluation and improvement based on typical V2X application at intersection. *Proceedings - 2017 Chinese Automation Congress, CAC 2017, 2017-Janua*, 556–561. doi: 10.1109/CAC.2017.8242830

Shrestha, R., Bajracharya, R., & Nam, S. Y. (2018). Challenges of Future VANET and Cloud-Based Approaches. *Wireless Communications and Mobile Computing*, 2018. doi: 10.1155/2018/5603518

Singh, J., Singh, P., Hedabou, M., & Kumar, N. (2023). An Efficient Machine Learning-based Resource Allocation Scheme for SDN-enabled Fog Computing Environment. , 1–15. doi: 10.1109/TVT.2023.3242585

Sodhro, A. H., Sodhro, G. H., Guizani, M., Pirbhulal, S., & Boukerche, A. (2020). AI-Enabled Reliable Channel Modeling Architecture for Fog Computing Vehicular Networks. *IEEE Wireless Communications*, 27(2), 14–21. doi: 10.1109/MWC.001.1900311

Sommer, C., Eckhoff, D., Brummer, A., Buse, D. S., Hagenauer, F., Joerer, S., & Segata, M. (2019). *Veins: The open source vehicular network simulation framework*. doi: 10.1007/978-3-030-12842-5_6

Sookhak, M., Yu, F. R., He, Y., Talebian, H., Safa, N. S., Zhao, N., . . . Kumar, N. (2017). Fog Vehicular Computing. *IEEE Vehicular technology magazine*(September), 2–11.

Tang, C., Wei, X., Zhu, C., Wang, Y., & Jia, W. (2020). Mobile Vehicles As Fog Nodes For Latency Optimization In Smart Cities. *IEEE Transactions on Vehicular Technology*, 14(8), 1–1. doi: 10.1109/tvt.2020.2970763

Tiku, M. L. (1967). Tables of the power of the f-test. *Journal of the American Statistical Association*, 62(318), 525-539. Retrieved from <https://doi.org/10.1080/01621459.1967.10482926> doi: 10.1080/01621459.1967.10482926

- Tiwari, R., & Kumar, N. (2015). Cooperative Gateway Cache Invalidation Scheme for Internet-Based Vehicular Ad Hoc Networks. *Wireless Personal Communications*, 85(4), 1789–1814. doi: 10.1007/s11277-015-2867-3
- Tiwari, R., & Kumar, N. (2016). An adaptive cache invalidation technique for wireless environments. *Telecommunication Systems*, 62(1). doi: 10.1007/s11235-015-0070-1
- Tiwari, R., Mittal, M., Garg, S., & Kumar, S. (2022). Energy-Aware Resource Scheduling in FoG Environment for IoT-Based Applications. *Lecture Notes on Data Engineering and Communications Technologies*, 74, 1–19. Retrieved from https://link.springer.com/chapter/10.1007/978-981-16-3448-2_1 doi: 10.1007/978-981-16-3448-2_1
- Tiwari, R., Sharma, H. K., Upadhyay, S., Sachan, S., & Sharma, A. (2019). Automated parking system-cloud and IoT based technique. *International Journal of Engineering and Advanced Technology*, 8(4C), 116–123. doi: 10.13140/RG.2.2.26946.09929
- Vaquero, L. M., & Rodero-Merino, L. (2014). Finding your way in the fog: Towards a comprehensive definition of fog computing. *Computer Communication Review*, 44(5), 27–32. doi: 10.1145/2677046.2677052
- Varghese, A., & Tandur, D. (2014). Wireless requirements and challenges in Industry 4.0. In *Proceedings of 2014 international conference on contemporary computing and informatics, ic3i 2014*. doi: 10.1109/IC3I.2014.7019732
- Vergis, S., Komianos, V., Tsoumanis, G., Tsipis, A., & Oikonomou, K. (2020). A Low-Cost Vehicular Traffic Monitoring System Using Fog Computing. *Smart Cities*, 3(1), 138–156. doi: 10.3390/smartcities3010008
- Vohra, K., Pal, G., Kumar Gupta, V., Singh, S., Bansal, Y., & Author, C. (2011). a I O Ce Asiatica Li : a Review O Rece Research. *Pharmacologyonline*, 2(July), 440–462.
- Wang, Y., Xu, C., Zhou, Z., Pervaiz, H., & Mumtaz, S. (2018). Contract-Based Resource Allocation for Low-Latency Vehicular Fog Computing. *IEEE International Symposium on Personal, Indoor and Mobile Radio Communications, PIMRC, 2018-Septe*, 812–816. doi: 10.1109/PIMRC.2018.8580843

- Wang, Z., Zheng, S., Ge, Q., & Li, K. (2020). Online Offloading Scheduling and Resource Allocation Algorithms for Vehicular Edge Computing System. *IEEE Access*, 8, 52428–52442. doi: 10.1109/ACCESS.2020.2981045
- Wang, Z., Zhong, Z., & Ni, M. (2018). Application-aware offloading policy using SMDP in vehicular fog computing systems. In *2018 IEEE International Conference on Communications Workshops, ICC Workshops 2018 - Proceedings*. doi: 10.1109/ICCW.2018.8403696
- Wei, Z., Li, B., & Zhang, R. (2023). Many-to-Many Task Offloading in Vehicular Fog Computing : A Multi-Agent Deep Reinforcement Learning Approach. , *XX(Xx)*, 1–16. doi: 10.1109/TMC.2023.3250495
- Wu, Q., Liu, H., Wang, R., Fan, P., Fan, Q., & Li, Z. (2020). Delay-Sensitive Task Offloading in the 802.11p-Based Vehicular Fog Computing Systems. *IEEE Internet of Things Journal*, 7(1), 773–785. doi: 10.1109/JIOT.2019.2953047
- Xiao, X., Hou, X., Chen, X., Liu, C., & Li, Y. (2019). Quantitative analysis for capabilities of vehicular fog computing. *Information Sciences*, 501, 742–760. Retrieved from <https://doi.org/10.1016/j.ins.2019.03.065> doi: 10.1016/j.ins.2019.03.065
- Xiao, Y., & Zhu, C. (2017). Vehicular fog computing: Vision and challenges. *2017 IEEE International Conference on Pervasive Computing and Communications Workshops, PerCom Workshops 2017*, 6–9. doi: 10.1109/PERCOMW.2017.7917508
- Xu, X., Liu, K., Xiao, K., Feng, L., Wu, Z., & Guo, S. (2020). Vehicular Fog Computing Enabled Real-Time Collision Warning via Trajectory Calibration. *Mobile Networks and Applications*. doi: 10.1007/s11036-020-01591-7
- Xu, Z., Li, X., Zhao, X., Zhang, M. H., & Wang, Z. (2017). DSRC versus 4G-LTE for connected vehicle applications: A study on field experiments of vehicular communication performance. *Journal of Advanced Transportation*, 2017. doi: 10.1155/2017/2750452
- Yadav, R., Zhang, W., Kaiwartya, O., Song, H., & Yu, S. (2020). Energy-Latency Tradeoff for Dynamic Computation Offloading in Vehicular Fog Com-

puting. *IEEE Transactions on Vehicular Technology*, 69(12), 14198–14211. doi: 10.1109/TVT.2020.3040596

Zadobrischi, E., & Damian, M. (2021). Vehicular communications utility in road safety applications: A step toward self-aware intelligent traffic systems. *Symmetry*, 13(3), 1–22. doi: 10.3390/sym13030438

Zhang, J., Wang, F. Y., Wang, K., Lin, W. H., Xu, X., & Chen, C. (2011). Data-driven intelligent transportation systems: A survey. *IEEE Transactions on Intelligent Transportation Systems*, 12(4), 1624–1639. doi: 10.1109/TITS.2011.2158001

Zhang, K., Peng, M., & Sun, Y. (2020). Delay-Optimized Resource Allocation in Fog based Vehicular Networks. *IEEE Internet of Things Journal*, 1–1. doi: 10.1109/jiot.2020.3010861

Zhang, Y., Wang, C. Y., & Wei, H. Y. (2019). Parking Reservation Auction for Parked Vehicle Assistance in Vehicular Fog Computing. *IEEE Transactions on Vehicular Technology*, 68(4), 3126–3139. doi: 10.1109/TVT.2019.2899887

Zhou, S., Sun, Y., Jiang, Z., & Niu, Z. (2019). Exploiting Moving Intelligence: Delay-Optimized Computation Offloading in Vehicular Fog Networks. *IEEE Communications Magazine*. doi: 10.1109/MCOM.2019.1800230

Zhou, Z., Liao, H., Wang, X., Mumtaz, S., & Rodriguez, J. (2020). When Vehicular Fog Computing Meets Autonomous Driving: Computational Resource Management and Task Offloading. *IEEE Network*. doi: 10.1109/MNET.001.1900527

Zhu, C., Pastor, G., Xiao, Y., & Ylajaaski, A. (2018). Vehicular Fog Computing for Video Crowdsourcing: Applications, Feasibility, and Challenges. *IEEE Communications Magazine*. doi: 10.1109/MCOM.2018.1800116

Zhu, C., Tao, J., Pastor, G., Xiao, Y., Ji, Y., Zhou, Q., . . . Yla-Jaaski, A. (2019). Folo: Latency and quality optimized task allocation in vehicular fog computing. *IEEE Internet of Things Journal*. doi: 10.1109/JIOT.2018.2875520

Zhu, G., Liu, D., Du, Y., You, C., Zhang, J., & Huang, K. (2020). Toward an Intelligent Edge: Wireless Communication Meets Machine Learning. *IEEE Communications Magazine*, 58(1), 19–25. doi: 10.1109/MCOM.001.1900103

Curriculum Vitae

Deep Chandra Binwal received his bachelor's degree in Electronics and Telecommunication from the Institution of Engineers (India) in 2004, M.E in Communications and Networking from Madhav Institute of Technology and Science, Gwalior, India in 2006, PG Diploma in Aeronautical Engineering from Visveswaraya University, Belgaum, India in 2010, M Tech in Cyber Security from Defence Institute of Advanced Technology, Pune, India in 2018. He is a corporate member of the Institution of Engineers (India). He has more than 20 years of professional and teaching experience with the Indian Air Force in the areas of Wireless Communication, Computer Networks, and Cyber Security. He is currently pursuing his Ph.D. degree from the University of Petroleum and Energy Studies, Dehradun, India.

List of Publications

Journals:

1. D. C. Binwal and M. Kapoor, “A Survey on Architecture, Applications, and Challenges in Vehicular Fog Computing,” , Int. J. Sensors, Wirel. Commun. Control, vol. 12, 2022, doi: 10.2174/2210327912666220127130014. (SCOPUS), Published.
2. D. C. Binwal, R. Tiwari, and M. Kapoor, “Modelling and Optimization of Vehicular Fog Network Towards Minimizing Latency”, Journal of Mobile Networks and Applications, Springer. SCIE, Impact Factor (3.8 - 2022), Published. <https://doi.org/10.1007/s11036-023-02197-5>.
3. D. C. Binwal, R. Tiwari, and M. Kapoor, “ A Lightweight Energy- Efficient Technique for QoS Enhancement in Urban VFC for Intelligent Transportation System”, Journal of Network and Systems Management , Springer. SCIE, Impact Factor (3.6 - 2022), Published. <https://doi.org/10.1007/s10922-023-09759-8>.
4. D. C. Binwal, and R. Tiwari “ ML-Assisted Vehicular Fog Node Selection Technique for Latency and Energy Optimization ”. Submitted to: Journal of Mobile Networks and Applications, Springer. SCIE, Impact Factor (3.8 - 2022), (Under peer review).

Conferences :

1. D. C. Binwal, R. Tiwari, and M. Kapoor, “Vehicle as Fog Server in Intelligent Transportation System,” in AIR 2022, International Conference on Advancement in Intradisciplinary Research, Published in Communications in Computer and Information Science, CCIS Volum., Springer, 2022, pp. 94–101. (SCOPUS Indexed), Published. <https://doi.org/10.1007/978-3-031-23724-9-9>.



Plagiarism Certificate

1. We **Dr. Shamik Tiwari** (Internal Guide), **Dr. Monit Kapoor** (Co Guide/ External Guide) certify that the Thesis titled **A Real-time Dynamic Fog Node Selection Technique to Enhance QoS and Reduce Energy Consumption in Vehicular Fog Network** submitted by Scholar **Mr. Deep Chandra Binwal** having **SAP ID 500078646** has been run through a Plagiarism Check Software and the Plagiarism Percentage is reported to be **8 %**.
2. The Plagiarism Report generated by the Plagiarism Software is attached.

Signature of the Internal Guide

Signature of External/Co-Guide

Signature of the Scholar

deepthesis

by Deep Deep

Submission date: 09-May-2024 10:31AM (UTC+0530)

Submission ID: 2374864760

File name: oS_and_Reduce_Energy_Consumption_in_Vehicular_Fog_Network_2.pdf (48.31M)

Word count: 34481

Character count: 192387

deepthesis

ORIGINALITY REPORT

8%

SIMILARITY INDEX

5%

INTERNET SOURCES

5%

PUBLICATIONS

1%

STUDENT PAPERS

MATCH ALL SOURCES (ONLY SELECTED SOURCE PRINTED)

2%

★ www.researchgate.net

Internet Source

Exclude quotes Off

Exclude bibliography On

Exclude matches Off