

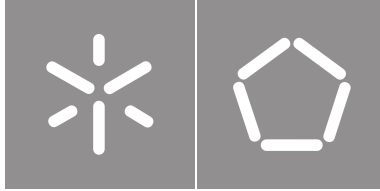


Universidade do Minho
Escola de Engenharia

Elídio Tomás da Silva

Context-Aware Routing Model for NDN-based VANET

Context-Aware Routing Model for NDN-based VANET
Elídio da Silva



Universidade do Minho

Escola de Engenharia

Elídio Tomás da Silva

Context-Aware Routing Model for NDN-based VANET

Doctorate Thesis

Doctorate in Electronics and Computer Engineering

Specialization in Information Systems and Computer

Communications

Work developed under the supervision of:

Prof. Doutor Joaquim Melo H. de Macedo

Prof. Doutor António Luís Duarte Costa

COPYRIGHT AND TERMS OF USE OF THIS WORK BY A THIRD PARTY

This is academic work that can be used by third parties as long as internationally accepted rules and good practices regarding copyright and related rights are respected.

Accordingly, this work may be used under the license provided below.

If the user needs permission to make use of the work under conditions not provided for in the indicated licensing, they should contact the author through the RepositoriUM of Universidade do Minho.

License granted to the users of this work



Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International
CC BY-NC-SA 4.0

<https://creativecommons.org/licenses/by-nc-sa/4.0/deed.en>

Acknowledgements

I would like to begin by expressing my deepest gratitude to my supervisors, Professor Joaquim Macedo and Professor António Costa. It is difficult to fully articulate my appreciation for your guidance throughout this entire journey, culminating in this thesis. I sincerely thank you for your time, patience, encouragement, and unwavering support — even during moments of personal difficulty. The knowledge and values I have learned from you will remain with me and guide my future academic and professional path.

I am also grateful to the University of Minho, particularly the ALGORITMI Research Centre, for providing the opportunity and the conditions necessary for the development of this work.

My sincere appreciation goes to Universidade Lúrio (UniLúrio) for the institutional support and for granting me the opportunity to pursue this doctoral degree. I extend special thanks to Professor Francisco Noa for his encouragement and support.

To my beloved wife, Josefina, and to my children, Albertina, Tomás, and Elídio Jr., I owe more than words can express. I recognize that no achievement can fully compensate for the psychological, social, and emotional burden caused by my absences — even when I was physically present but mentally absorbed in this work. Josefina, you have endured more than I could have imagined, raising our children and managing everything with strength and resilience. Your unconditional support and unwavering trust in me sustained me during moments when my own confidence faltered. This accomplishment is as much yours as it is mine.

In loving memory of my mother, whose values, sacrifices, and love continue to guide me. To my father, Tomás Silva, and to my brothers — Nico, Adélia, the eldest Alberto Bila, José, Felizardo, Clésio, and Hermenegildo — as well as to my nephews and nieces, I express my heartfelt gratitude. Your support and trust have been invaluable throughout this journey.

To my friends, both in Portugal and in Mozambique — Santos, Simão, Ossifo, Igor, Aiúba, Capalo, Paulo, Celso, José, Júlio, Cristóvão, Loite, Nilton, Khotasse, Dércio, and Frederico — thank you for the moments of relaxation, encouragement, and friendship. The order is entirely random, but the gratitude is equally sincere.

Finally, this work was financially supported initially by the SSDAIP project at UniLúrio and subsequently by Fundação para a Ciência e a Tecnologia (FCT) through the doctoral grant PRT/BD/154352/2023.

STATEMENT OF INTEGRITY

I hereby declare having conducted this academic work with integrity. I confirm that I have not used plagiarism or any form of undue use of information or falsification of results along the process leading to its elaboration.

I further declare that I have fully acknowledged the Code of Ethical Conduct of the Universidade do Minho.

Resumo

Modelo de encaminhamento sensível ao contexto para VANET baseada em NDN

As cidades inteligentes, que são já uma realidade nos países desenvolvidos, estão também a tornar-se comuns nos países em desenvolvimento. Um componente essencial das cidades inteligentes é o sistema de transportes, vulgarmente designado por Sistemas de Transporte Inteligentes (ITS). As Redes Ad-hoc Veiculares (VANET) são componentes essenciais dos ITS e são responsáveis por permitir sistemas de transporte melhorados ao suportar aplicações avançadas, como a gestão de tráfego em tempo real, a resposta a emergências, a coordenação de veículos autónomos e a eficiência geral do tráfego. No entanto, a sua topologia altamente dinâmica e conectividade intermitente representam desafios significativos para o encaminhamento de conteúdos eficiente e fiável. O principal desafio é a dificuldade de manter as tabelas de encaminhamento - Forwarding Information Base (FIB), actualizadas. Isto, e o facto de os requisitos de comunicação actuais serem centrados no conteúdo, fazem com que as abordagens tradicionais de disseminação de conteúdo baseadas em IP tenham dificuldade em lidar com estes desafios.

Esta tese explora a integração da Rede de Dados Nomeados (NDN), uma arquitectura de rede emergente centrada no conteúdo, e da VANET para resolver os problemas acima referidos. As soluções de última geração ultrapassam o problema da manutenção da FIB, evitando a sua gestão e recorrendo à chamada tabela de nós vizinhos (NT), da qual é seleccionado um próximo salto. Contudo, as NT também precisam de ser actualizadas. Para o conseguir, algumas soluções recorrem ao *broadcasting* de *beacons*. Esta tese propõe uma Estratégia de Encaminhamento baseado no contexto e Mobilidade para as V-NDN (CMAF) e um Modelo de Routing consciente da Mobilidade para as V-NDN (MARV), ambos utilizando a Previsão de Mobilidade (PM) de nós. CMAF e MARV aproveitam os pacotes escutados, no canal sem fios, para extrair informação de mobilidade do nó e utilizá-la para gerir e actualizar a NT e FIB. O modelo final visa reduzir o *Overhead* de Transmissão (TO), mantendo um elevado desempenho na disseminação de conteúdos.

Através de extensivas simulações o modelo proposto demonstra 1) superioridade em termos de Rácio de Satisfação de Interesse (ISR) em comparação com o estado da arte e 2) uma redução adicional de TO ao aplicar a PM para gerir a FIB e NT.

Palavras-chave: Encaminhamento, Previsão de mobilidade, Redes ad hoc Veiculares (VANET), Redes de Dados Nomeados (NDN), Routing.

Abstract

Context-Aware Routing Model for NDN-based VANET

Smart cities, which are already a reality in developed countries, are also becoming common in developing countries. An essential component of smart cities is the transportation system, commonly referred to as Intelligent Transportation System (ITS). Vehicular Ad-hoc Networks (VANETs) are an essential components of ITS and are responsible for enabling improved transportation systems by supporting advanced applications such as real-time traffic management, emergency response, autonomous vehicle coordination, and overall traffic efficiency and driving experience. However, its highly dynamic topology and intermittent connectivity pose significant challenges for efficient and reliable content routing. The main challenge is the difficulty of keeping routing tables - Forwarding Information Base (FIB), up to date. This, and the fact that today's communication requirements are content-centric, makes traditional IP-based content dissemination approaches struggle to address these challenges.

This thesis explores the integration of Named Data Networking (NDN), an emerging content-centric network architecture, and VANET to address the above issues. State-of-the-art solutions overcome the problem of maintaining FIB, by avoiding its management and resorting to the so-called Neighbor Table (NT), from which a next hop is selected. However, NTs also need to be updated. To achieve this, some solutions use beacon broadcasting. This thesis proposes a Context and Mobility-Based Routing (CMAF) strategy and a Mobility-Aware Routing Model for V-NDN (MARV), both using node Mobility Prediction (MP). CMAF and MARV leverage overheard packets, in the wireless channel, to extract node mobility information and use it to manage and update NT and FIB. The final model aims to reduce Transmission Overhead (TO), whilst maintaining high performance in content dissemination.

Through extensive simulations using realistic urban mobility routes the proposed model demonstrates 1) superiority in terms of Interest Satisfaction Ratio (ISR) compared to the state-of-the-art and 2) a further reduction of TO when applying MP to manage FIB and NT.

Keywords: Forwarding, Mobility prediction, Named Data Networking (NDN), Routing, Vehicular ad hoc Networks (VANET).

Contents

List of Figures	xi
List of Algorithms	xiii
Acronyms	xiv
1 Introduction	1
1.1 Overview and Problem Statement	1
1.2 Motivation	3
1.3 Aim and Research Questions	6
1.4 Assumptions	7
1.5 Research Methodology	8
1.6 Thesis Contributions	9
1.7 Thesis Outline	11
2 Theoretical Background	13
2.1 Overview of Vehicular Ad hoc Network (VANET)	13
2.1.1 Architecture	13
2.1.2 VANET proprieties	14
2.1.3 VANET applications	15
2.1.4 Security, Privacy and Trust	17
2.1.5 Main challenges and research issues	17
2.2 Overview of Information Centric Networking (ICN)	18
2.2.1 Architecture of NDN (a specific realization of ICN)	20
2.2.2 Routing and Forwarding in NDN	22
2.2.3 NDN In-network Caching	25
2.2.4 NDN Built-in Security	25
2.3 VANET enhanced by NDN	26

2.4	Mobility Prediction in VANET	26
2.4.1	Kalman Filter (KF)	29
2.4.2	Extended Kalman Filter (EKF)	31
2.4.3	Hidden Markov Model (HMM)	32
2.5	Summary	35
3	NDN-based VANET - State-of-the-art	36
3.1	Context	36
3.1.1	Objectives	37
3.1.2	Main contributions	37
3.2	Related surveys	38
3.3	Methodology	40
3.4	Research Questions and Search Strings	40
3.5	Selected sources and studies	41
3.6	Inclusion and exclusion criteria	42
3.7	Inclusion of primary studies	42
3.8	Data extraction and synthesis	46
3.8.1	Research Question (RQ1) - Routing	46
3.8.2	Research Question (RQ2) - Caching	51
3.8.3	Research Question (RQ3) - Security, Privacy and Trust	57
3.9	Data analysis and Results Interpretation	59
3.9.1	Research Question (RQ1) - Routing	59
3.9.2	Research Question (RQ2) - Caching	64
3.9.3	Research Question (RQ3) - Security, Privacy and Trust	70
3.10	Improving VANETs using NDN	71
3.10.1	Research Question (RQ1) - Routing	71
3.10.2	Research Question (RQ2) - Caching	72
3.10.3	Research Question (RQ3) - Security, Privacy and Trust	72
3.11	Forwarding solutions	73
3.11.1	Sender-Oriented Approaches—Neighborhood-Aware	73
3.11.2	Receiver-Oriented Approaches	75
3.11.3	Comparison of the Selected State-of-the-Art Solutions	79
3.12	Summary	81
4	Design Overview	82
4.1	Context	82
4.2	Realization of NDN-Based VANET	83
4.3	NDN Forwarding Plane: Context and Mobility-Aware Forwarding Model for V-NDN (CMAF)	84

4.3.1	Novelty of the Proposed Forwarding Model	84
4.3.2	Overview of Cached Content and Neighbor Discovery Process	85
4.3.3	Processing of Incoming Data	89
4.3.4	Procedure for Incoming Interest	92
4.4	NDN Routing Plane: Mobility-Aware Routing Model for V-NDN (MARV)	96
4.4.1	Novelty of the Proposed Routing Model	97
4.4.2	MARV Architecture Description	97
4.4.3	Initial Route Discovery	99
4.4.4	FIB's Management Procedure	100
4.5	Short-Term Mobility Prediction (STMP)	100
4.6	Long-term mobility prediction (LTMP)	101
4.7	Summary	105
5	Implementation of the Proposed Model	106
5.1	Network Simulator	106
5.2	Road Traffic Simulator	108
5.3	Technical specifications	110
5.3.1	Hardware	110
5.3.2	Software	110
5.4	System Implementation	111
5.4.1	Main NDN Structures Enhancements	112
5.4.2	Procedure for Modifying the Forwarder	115
5.4.3	Procedure for Adding new Forwarding Strategy	115
5.4.4	Procedure for Adding new Routing protocol	117
5.4.5	Additional Applications and Classes Included in the Simulator	117
5.5	Main Challenges on Implementing the Model	118
5.6	Summary	119
6	NDN-based VANET: Performance Evaluation	121
6.1	Selected Metrics	121
6.2	Simulation Environment	122
6.3	Simulations results for CMAF	124
6.3.1	Simulations Using the Road Scenario from the City of Porto	124
6.3.2	Simulations Using a Manhattan-Like Road Network	128
6.3.3	Discussion at the Data Plane Level	129
6.4	Simulations results for MARV	130
6.4.1	Discussion at the Control Plane Level	132
6.5	General Discussion	133

6.6	Summary	134
7	Conclusions and Future Work	135
7.1	Summary	135
7.2	Conclusions and Answers to the Research Questions	136
7.3	Main Limitations and Recommendations	138
7.4	Future Work	139
	Bibliography	141

List of Figures

1	The VANET architectures	14
2	Network-based classification of VANET applications	16
3	NDN packets. Interest (left) and Data (right).	21
4	The forwarding procedure in NDN architecture.	23
5	Illustration of the routing procedure.	24
6	Routing and forwarding planes in a NDN router.	24
7	The Hidden Markov Model	33
8	Systematic literature review process.	40
9	Year of publication of the selected studies, per area	45
10	Communication model for the proposed routing schemes	60
11	Operation mode of the proposed routing schemes.	60
12	Distribution of the selected studies in terms of the proposed caching mode.	64
13	Distribution of the selected studies in terms of the communication model.	65
14	The surveyed main security and privacy attacks in NDN-based VANET.	70
15	Beacon processing in intermediate nodes	88
16	High level overview of Data processing in the intermediate nodes	92
17	High level overview of Interest processing in the intermediate nodes	94
18	Hypothetical user's mobility trace	103
19	Structure of the ndnSIM.	107
20	Manhattan-like scenario.	109
21	Scenario based on the City of Porto, Portugal.	109
22	Implementation of the ndnSIM	112
23	Structure of the PIT.	114
24	Structure of FIB with additional fields for NMSI for each next-hop.	115
25	ISR and TO, for the selected protocols, without Data hop-limit.	125

26	ISD and jitter, without limiting Data hop count.	125
27	Average packet retransmissions and hop count presented by the protocols.	126
28	ISR and TO, with limited Data forwarding hop-count.	126
29	ISD and jitter, for simulations with limited Data hop-count	127
30	ISR and TO, where Interest and Data are only allowed for limited hops.	127
31	ISD and jitter, for simulations with limited Interest and Data hop-counts	128
32	Average packet retransmissions and hop count.	128
33	ISD and TO, for the Manhattan-like grid.	129
34	ISD and jitter for the selected protocols, for the Manhattan-like grid.	129
35	Average retransmissions and hop count presented by the protocols.	129
36	ISR and TO, for the selected protocols, without Data hop-limit.	131
37	Average number of transmitted Interest and Data	131
38	ISD and jitter, without limiting Data hop count.	132
39	Average packet retransmissions and hop count presented by the protocols.	132

List of Algorithms

4.1	The NT insert or update procedure	86
4.2	Incoming Data processing	89
4.3	Incoming Interest processing	93
4.4	Routing process	99
4.5	The STMP algorithm, based on EKF	102
4.6	The LTMP algorithm, based on HMM	105

Acronyms

ANN	Artificial Neural Network (<i>pp. 9, 27–29</i>)
AP	Access Point (<i>pp. 14, 51, 53, 54, 65, 67, 68, 72</i>)
API	Application Programming Interface (<i>pp. 19, 20</i>)
AT	Address Table (<i>p. 54</i>)
BF	Bloom Filter (<i>pp. 10, 20, 48, 61, 63, 85, 87, 88, 95, 116, 117, 133</i>)
BN	Bayesian Networks (<i>p. 9</i>)
BS	Base Station (<i>pp. 14, 51, 54, 65, 67–69, 72</i>)
BSA	Basic Set of Applications (<i>p. 15</i>)
CACS	Comprehensive Automobile Traffic Control System (<i>p. 1</i>)
CCLF	Content Connectivity and Location-Aware Forwarding (<i>pp. 76, 121, 124, 125, 128</i>)
CCN	Content Centric Network (<i>pp. 2, 19, 20, 36, 38, 39, 41, 47–50, 83, 106</i>)
CCT	Cached Content Table (<i>pp. 88, 91, 94–96, 116</i>)
CH	Cluster Head (<i>pp. 48, 49, 53, 62, 63, 65, 68</i>)
CHR	Cache Hit Ratio (<i>p. 64</i>)
CMAF	Context and Mobility-Aware Forwarding Model for V-NDN (<i>pp. 11, 12, 79, 80, 84, 89–92, 94–96, 99–101, 105, 109, 116–118, 120, 122–136, 139</i>)
CN	Content Name (<i>pp. 21–23, 50, 62, 75, 79, 140</i>)
CPA	Cache Pollution/Poisoning Attack (<i>pp. 57, 58</i>)
CPMT	Content Popularity Monitor Table (<i>p. 55</i>)
CS	Content Store (<i>pp. 7, 9, 10, 21–24, 50, 51, 55, 56, 58, 63, 65–67, 69, 72, 74, 78, 80, 85, 87–91, 93, 95, 106, 116, 117, 120, 124, 133–135, 138</i>)
DDoS	Distributed Denial of Service (<i>p. 70</i>)
DIF	Distributed IPC Facility (<i>p. 20</i>)
DM	Data Mining (<i>p. 9</i>)

DONA	Data-Oriented Network Architecture (pp. 19, 20)
DoS	Denial of Service (p. 4)
DR	Dead Reckoning (pp. 27, 28)
DSRC	Dedicated Short Range Communication (p. 63)
DSRM	Design Science Research Methodology (p. 8)
DTN	Delay Tolerant Networks (pp. 38, 83, 91, 98)
EKF	Extended Kalman Filter (pp. 9, 28, 29, 31, 101)
ETSI	European Telecommunications Standards Institute (p. 15)
FC	Floating Content (p. 72)
FIB	Forwarding Information Base (pp. 5, 10, 21–26, 37, 46–50, 59, 61–63, 71, 73, 79, 82, 83, 89, 91, 94, 96–101, 105, 107, 114, 115, 117, 119, 124, 130–137, 139)
FIFO	First In First Out (p. 66)
FIIA	False Information Injection Attack (pp. 57, 70)
GPS	Global Position System (pp. 2, 7, 14, 27, 46, 98)
GS	Google Scholar (pp. 41–45)
HBFR	Hierarchical BF Routing (p. 48)
HMM	Hidden Markov Model (pp. 3, 9, 27–29, 32–34, 54, 56, 68, 69, 101–104)
ICN	Information Centric Networking (pp. 2, 4, 5, 13, 19, 36, 38, 40, 41, 48, 54, 62, 68, 83, 106)
IFA	Interest Flooding Attack (pp. 57–59, 70)
ILP	Integer Linear Programming (p. 51)
IoT	Internet of Things (p. 90)
IoV	Internet of Vehicles (pp. 41, 55, 56, 69)
IP	Internet Protocol (pp. 1–6, 8, 9, 18–20, 22, 24–26, 35, 36, 39, 50, 57, 62, 63, 70, 71)
IPC	Inter-Process Communication (p. 20)
ISD	Interest Satisfaction Delay (pp. 119, 121, 125, 127–136)
ISR	Interest Satisfaction Ratio (pp. 75, 77–79, 121, 124–131, 133, 135, 136)
ITS	Intelligent Transportation System (pp. 1, 3, 6, 13, 35)
IVC	Inter Vehicular Communication (p. 48)
KF	Kalman Filter (pp. 3, 28–31, 120)
LET	Link Expiration Time (pp. 48, 73, 74)

LFU	Least Frequently Used (<i>pp. 66, 67</i>)
LPM	Longest Prefix Match (<i>p. 23</i>)
LRU	Least Recently Used (<i>pp. 66–69</i>)
LSTM	Long-Short-Term Memory (<i>p. 68</i>)
LTE	Long-Term Evolution (<i>p. 68</i>)
LTMP	Long-Term Mobility Prediction (<i>pp. 3, 7, 11, 82, 100, 101, 105, 117, 119, 120, 135, 136</i>)
MA	Mobile Agent (<i>p. 63</i>)
MAC	Media Access Control (<i>pp. 46, 62, 74, 99, 100, 111, 113, 114, 119</i>)
MANET	Mobile Ad hoc NETWORKS (<i>pp. 1, 13, 37, 38, 42, 47, 108, 111</i>)
MARV	Mobility-Aware Routing Model for V-NDN (<i>pp. 12, 91, 97, 98, 100, 105, 109, 115, 117, 119, 122, 130–135, 139</i>)
MC	Markov Chain (<i>pp. 9, 27–29</i>)
MES	Mobile Edge Server (<i>pp. 49, 63, 69</i>)
ML	Machine Learning (<i>p. 28</i>)
MM	Markov Model (<i>pp. 32, 53, 67, 68</i>)
MPI	Message Passing Interface (<i>p. 119</i>)
NACK	Negative Acknowledgment (<i>pp. 22, 23</i>)
NDN	Named Data Networking (<i>pp. 2–6, 8–11, 13, 19–26, 35–39, 41, 42, 45, 46, 48, 50, 55–57, 59, 62, 64, 70, 71, 73, 74, 76, 79, 81–85, 89, 91, 95–97, 99, 106, 107, 111, 112, 114, 122, 130, 135, 140</i>)
NDNLP	Named Data Networking Link Protocol (<i>pp. 112, 113</i>)
ndnSIM	NDN Simulator (<i>pp. 9, 106–108, 110, 111, 115, 117–119, 122, 139</i>)
NDO	Named Data Object (<i>pp. 5, 19, 20</i>)
NetInf	Network Information (<i>pp. 19, 20</i>)
NFD	NDN Forwarder Daemon (<i>pp. 21, 106, 107, 112, 115, 117</i>)
NMSI	Node Mobility Status Information (<i>pp. 32, 84, 85, 92, 99, 100, 104, 105, 112, 113, 115–117, 119, 120, 134, 136</i>)
NRS	Named Resolution Service (<i>pp. 19, 20</i>)
NS-3	Network Simulator 3 (<i>pp. 106–108, 111, 119, 122</i>)
NT	Neighborhood Table (<i>pp. 50, 75, 76, 79, 80, 83–87, 91, 92, 94, 100, 101, 105, 116, 117, 123, 133–138</i>)
NTRsu	Neighborhood Table of RSUs (<i>pp. 86, 93, 94, 99, 116, 137</i>)
OSM	Open Street Map (<i>p. 108</i>)

PDR	Packet Delivery Ratio (<i>p. 121</i>)
PF	Particle Filters (<i>pp. 28, 29, 31</i>)
PIT	Pending Interest Table (<i>pp. 5, 21–23, 46, 50, 56, 76, 79, 89–92, 99, 114, 115, 117, 119, 132</i>)
PKI	Public Key Infrastructure (<i>pp. 17, 20</i>)
PoI	Point-of-Interest (<i>pp. 74, 79</i>)
PSIRP	Publish-Subscribe Internet Routing Paradigm (<i>pp. 19, 20</i>)
PURSUIT	Publish-Subscribe Internet Technologies (<i>p. 19</i>)
PVA	Privacy Violation Attack (<i>pp. 57, 59</i>)
QoE	Quality of Experience (<i>pp. 27, 66, 67, 95</i>)
QoS	Quality of Service (<i>pp. 4, 27, 95</i>)
RH	Resolution Handler (<i>pp. 19, 20</i>)
RINA	Recursive InterNetwork Architecture (<i>pp. 19, 20</i>)
RL	Residual Lifetime (<i>pp. 55, 69</i>)
RoI	Region of Interest (<i>p. 16</i>)
ROOF	Real-time Onsite Operations Facilitation (<i>p. 56</i>)
RQ	Research Question (<i>pp. 7, 40–46, 51, 57, 63, 69–71, 98, 136–138</i>)
RSU	Road Side Unit (<i>pp. 1, 3, 7, 10, 14, 46, 49–51, 53–56, 58, 59, 61–63, 65–70, 72, 79, 84–88, 93–96, 99, 101, 116–119, 124, 135, 137, 138</i>)
RTT	Round Trip Time (<i>pp. 22, 91, 107, 121</i>)
SDN	Software Defined Networking (<i>pp. 56, 59, 67</i>)
SLR	Systematic Literature Review (<i>pp. 3, 8, 11, 36, 37, 40, 81</i>)
SP	Staying Place (<i>p. 32</i>)
STMP	Short-Term Mobility Prediction (<i>pp. 3, 7, 11, 82, 86, 87, 100, 101, 105, 117, 119, 120, 124, 131, 135, 136, 139</i>)
SUMO	Simulation of Urban MObility (<i>pp. 106, 108–110, 122</i>)
TCP	Transmission Control Protocol (<i>pp. 4, 9, 19, 20, 26, 39</i>)
TLV	Type-Length-Value (<i>p. 113</i>)
TO	Transmission Overhead (<i>pp. 6, 10, 97, 122, 124–131, 133–136, 138, 139</i>)
TTL	Time To Live (<i>p. 49</i>)
UKF	Unscented Kalman Filter (<i>pp. 28, 29, 31</i>)
V-NDN	Vehicular Named Data Networking (<i>pp. 38, 68, 76, 121, 124, 125, 128</i>)

V2I	Vehicle to Infrastructure (<i>pp.</i> 14, 16, 41, 46, 47, 52, 59, 62, 63, 65, 67, 72, 79, 81, 84, 85, 96, 99, 119, 121, 124, 126, 127, 129, 130, 133, 135, 137, 138)
V2V	Vehicle to Vehicle (<i>pp.</i> 3, 13, 41, 46, 47, 52, 59, 62, 65, 67, 72, 81, 84, 85, 89, 90, 96, 99, 121, 124–127, 129–131, 137, 138)
V2X	Vehicle to Everything (<i>pp.</i> 14, 16, 41, 63, 72, 102)
VANET	Vehicular Ad hoc NETwork (<i>pp.</i> 1–11, 13–15, 17, 18, 20, 25–28, 35–38, 40–42, 45–47, 50, 53, 55, 57, 59–61, 63–65, 67, 70–73, 81–84, 91, 95–98, 101–103, 108, 117, 122, 124, 135, 137, 139, 140)
VCCN	Vehicular Content Centric Networks (<i>p.</i> 38)
VN	Virtual Node (<i>pp.</i> 50, 72)
WoS	Web of Science (<i>pp.</i> 41–45)
WSN	Wireless Sensor Networks (<i>p.</i> 38)

Introduction

This chapter presents a brief overview of the main concepts that make the ground for the development of the present thesis, followed by the problem statement. Then, the motivation, the aim and the research questions, along with the methodology that guided the development of this work are presented. Finally, the main contributions and the outline of this thesis are presented.

1.1 Overview and Problem Statement

Intelligent Transportation System (ITS) is a technological concept that can be thought of as the integration of information, communication technologies and applications. It aims at improving the performance of transport systems, by helping to manage its infrastructure and helps users to obtain an overall satisfaction with it [198]. This is not a new concept as since the 1970's a similar system existed in Japan - Comprehensive Automobile Traffic Control System (CACS) [176, 244].

CACS aimed at: a) reducing the road traffic congestion; b) reducing the exhaust fumes caused by traffic congestion, for green environment; c) preventing traffic accidents; and d) enhancing the public and social role of automobiles, by linking the communication to a central processing unit, thus constructing an integrated information system that could provide effective tools for road traffic management and control [134].

Vehicular Ad hoc NETWORK (VANET), a particular case of Mobile Ad hoc NETWORKS (MANET) [66], is a specific area of networking that has been under intensive research due to, in one part, its foreseen importance for the development of ITS [16]. Communication in VANETs is performed between vehicles and other capable wireless devices (e.g., Road Side Unit (RSU), Pedestrian smartphones and other vehicles). Each one of these wireless devices/nodes takes a role as an end-user device and wireless router to support a wider range of communications. VANET presents specific challenges aside the generic challenges presented by MANET, due to its characteristics which includes the frequent network partitioning, highly dynamic topology and short-lived links between nodes. Routing in VANET is one of these differences and challenges, which directly results from the aforementioned characteristics. VANET challenges are intimately related to the underlying network architecture, which currently is the Internet Protocol (IP)-based

network.

The IP-based network is host-centric. Current network communications, however, are content-centric by nature. That is, the key element for communication is the content itself, rather than the host. This shift in the current network communications poses new challenges to the current IP-based architecture. For instance, IP-based Internet's best-effort approach became inadequate for real-time applications, such as video-conferencing [211]. These factors stimulated the development of new architectural paradigms capable of natively supporting the new reality. The aforementioned new architectural paradigms are grouped under the so-called Information Centric Networking (ICN). Former and current ICN-based projects including their standardization are comprehensively detailed by [9, 276, 289]. Among these projects, Named Data Networking (NDN), a particular implementation of Content Centric Network (CCN) [118] have been considered as the most promising solution [301] to solve the aforementioned IP-based network issues.

NDN is a content-centric and name-based architecture. The contents identification in NDN is performed by names instead of IP addresses or host names. Identifying contents by names provides their dissociation from their location, application, storage, and means of transportation [151]. This propriety results in the advantage of being able to ubiquitously store and share content among several nodes, benefiting the mobility and other communication requirements of VANET applications and services. Furthermore, communications in NDN are session-less. That is, there is no need for a session-based communication and continuous channel availability as in the case of IP-base networks. This is one of the characteristics that combined with the use of NDN in-network caching leads to a time-decoupling design where a content producer/holder is not required to be online during the time a content is transferred to the consumer/requester [140].

Considering the aforementioned main characteristics of NDN (i.e., the hierarchical content naming scheme, session-less design and support for in-networking caching) and the fact that most of the communications in VANET follows a content-centric approach [251] - applications care more on accessing a particular service rather than the node offering such service [50] - NDN has been seen as a highly suitable architectural model, adequate to mitigate the IP-related issues in VANET. As such, the realization of VANET by NDN have been proposed by many researchers.

Integration of VANET and NDN is currently performed by designing clean application (e.g., [172, 194]), customizing new naming (e.g., [260, 277]) and caching schemes (e.g., [59, 183]), designing or adapting existing forwarding strategies (e.g., [69, 295]), or by introducing modifications into the NDN core architecture (e.g., [227, 185]) or even proposing new architectural guidelines for adapting NDN for VANET [112, 140]. Some of the specific areas where leveraging the synergies from both architectures (i.e., NDN, VANET) could be of an immense importance, are: routing and forwarding. Indeed, the realization of NDN-based VANET has ultimately seen an immense development. However, even this integration introduces several issues and challenges that still lack effective solutions.

In addition to adopting NDN, the widespread use of localization systems (e.g., Global Position System (GPS)) has been seen as an important mechanism for improving routing in VANET and overcome some

of the issues imposed by the intermittency of the links, which actually makes it difficult to maintain the created routes among nodes. The use of localization systems for tracking the geographical position and trajectory of the vehicles, has opened opportunities for the development of various mechanisms for predicting the mobility of vehicles on the road. Resorting to these mechanisms, a routing protocol may be able to more efficiently track the routes of content sources and better select the relay node with the highest probability of forwarding the content to the destination.

A Systematic Literature Review (SLR) performed in [232] and presented in Section 3, concluded that a still negligible number of solutions was proposed for NDN-based routing in VANET. Besides, the identified studies do not take advantage of the existing knowledge on mobility prediction, in order to improve routing. Furthermore, the proposed solutions do not leverage the in-network caching for better routing decision. We highlight that storage and computational resources are not restraints in VANET, thus their usage can more freely be explored.

This thesis aims at exploring and presenting solutions for some of the challenges and issues encountered on developing a NDN-based VANET. More precisely, the thesis proposes developing a context-aware forwarding and routing model for NDN-based VANET, enhanced by mobility prediction and caching. The proposed forwarding and routing model leverages in-network caching, and tracks the cached content in all nodes that intervene in forwarding (this process is explained later in Section 4.3.2.2 and Section 4.3.4.1). Additionally, the nodes take advantage of the existing RSU to improve the overall network performance. This is done considering that RSU have a wider knowledge of the network.

Several state-of-the-art solutions resort to beacon broadcasting for discovering neighbors and querying content sources in the network. For avoiding frequent beacon broadcast, this thesis proposes: 1) the inclusion of mobility information on all packets, for neighborhood management and 2) the inclusion of two mobility prediction schemes: a Long-Term Mobility Prediction (LTMP), based on Hidden Markov Model (HMM) and deployed in all RSU in the network and, in order to reduce the processing overhead, a lighter Short-Term Mobility Prediction (STMP) scheme, based on Kalman Filter (KF) in all vehicles.

1.2 Motivation

With the crescent automobile development, high speed vehicles are a crescent reality. Besides, autonomous vehicles, are in a high demand in this current reality aiming at making daily life more secure and convenient [191]. These vehicles are commonly found congesting roads in major cities worldwide. It is easy to find these vehicles jammed due to accidents or even small road maintenance. Timely efficient watchful suggestions from neighbor vehicles could be of a vital importance to improve traffic efficiency and the safety of the drivers and passengers [216]. In fact, the main purpose of VANET and these emerging new urban technologies is to provide ubiquitous connectivity to mobile users, safe and efficient Vehicle to Vehicle (V2V) communications in order to enable ITS [177]

As previously mentioned in Section 1.1, IP-based networks are becoming challenged by the needs of

the current network applications, which follow the content-centric approach. Some of the issues presented by the current IP-based networks, which result from the design at the network protocol architecture level, include [117]:

1. Naming, addressing and routing - which include, a) the lacking of application names that would render applications location-independent and facilitate, for instance, mobility and b) naming the interface instead of the node, which makes the support of multi-homing infeasible;
2. Mobility and multi-homing - which include specific issues such as, a) the need for specialized protocols to support multi-homing, and b) the need for specialized protocols to support mobility;
3. Quality of Service (QoS), resource allocation, and congestion control - which include the fact that a) there exists only end-to-end congestion control loops - which maximizes the length of the congestion control loop, and consequently maximizes the time to react to congestion, and b) the predatory congestion control - meaning that the protocol uses the lost or duplicate packets as a signal of congestion. This may cause the protocol to interpret packet loss due to link failure as congestion; and
4. Security (e.g., pervasive security problems, such as spam, Denial of Service (DoS) attacks, phishing [211]) - which due to the lack of application names, protocols are forced to expose the network addresses to applications. In addition, the absence of a generic security model forces most protocols to have their own security model, which makes the entire protocol stack complex and susceptible to security breaches.

ICN-based networks are proposed to solve, by design, the aforementioned issues. In summary,

1. Differently from Transmission Control Protocol (TCP)/IP protocol stack, where security and privacy is provided as extensions, in the ICN-approaches it is provided by design;
2. Whilst TCP/IP-based networks require a session for exchanging content between end-nodes, ICN-approaches are session-less and natively support asynchronous data exchange between nodes;
3. Interest aggregation in ICN-approaches provides a native support for efficient multicast data delivery;
4. Whereas the content is bound to its location in the IP-based networks, the self-consistency of content in ICN-approaches provides the ability for in-network caching.

The aforementioned properties of the ICN-based approaches are, in part, the reasoning behind the development of ICN-based VANET.

Among the proposed ICN-based architectures (see Section 2.2), NDN is chosen based on the comparison presented in Table 2, Section 2.2. In summary, NDN is chosen to the detriment of other approaches,

given that 1) it is one of the most researched architecture, among the other ICN-based approaches; 2) uses the hierarchical namespace scheme, which favors the scalability and aggregation; 3) the Named Data Object (NDO) granularity - in packets rather than objects, favors a better utilization of caching; 4) the routing NDO request approach is name-based and does not requires an additional service for name resolution.

Unicast and anycast communications between vehicles and servers that collect traffic informations from vehicles are essential and needed in VANET [50]. Usually, unicast based routing models resort to periodic beacon broadcast in order to: a) request prefix announcements from content sources; b) discover new content sources; and c) track the node's neighbors. VANET is essentially based on wireless communications. In this network, when a node sends a given packet, it is overheard by all the surrounding nodes within the sender's communication range. This thesis proposes a model that takes advantage of the wireless channel propriety and instead of deliberately broadcasting a packet (beacon) for tracking the node's neighbors, for instance, it appends some control information on each packet, thus each received packet serve as a beacon, bringing important information for tracking vehicles' routes. The appended control information is extracted on each node receiving the overheard packet. This way the node learns about all of its neighbors. The inclusion of the control information on each packet is complemented by the modification of other NDN structures such be, the Forwarding Information Base (FIB), Pending Interest Table (PIT) and the inclusion of additional specific tables, as explained later. Furthermore, to avoid broadcast and be able to send a packet to specific node in VANET (unicast dissemination), an additional mechanism to identify the node is necessary. The MAC address is chosen for node identification, see Section 4.4.3, and is included on the appended control information.

Several proposals for forwarding strategies in NDN for VANET are based on flooding, and on a particular scheme to control the rebroadcast, e.g., resorting to distances to the content provider, and timers for deferring the subsequent broadcast. This mechanism has the advantage of simplicity, does not require the knowledge about the neighbor nodes or even the usage of FIB. However, it generally result in problems that include traffic congestion, packet collisions and delivery delays due to the broadcast storm problem. Some representative proposals using this mechanism include studies by [97, 199]. A comprehensive list of similar studies can be found, for instance, in [140].

A relatively more intelligent alternative for broadcast and the usage of defer timers and distances is the identification and selection of possible relay nodes, by using unicast communication model. This approach requires an updated knowledge about the node's neighbors and the geographical location of other intermediate nodes, from which the better relay between the current node and the content source can be selected. This mechanism can reduce the traffic congestion, collision and delivery delays, at a cost of overhead related to the maintenance of an updated list of neighbors and tracking of other nodes in the network. Moreover, a survey on routing protocols based on multimetrics [246], uses metrics such as *distance*, *density*, *link stability*, *speed* and *position* to propose a better solution on choosing a routing protocol for a specific model of VANET. They conclude that protocols based on *speed*, *link stability* and *distance* are the most promising and versatile in VANET. The conclusion refers to the IP-based VANET

but hypothetically the same is true for NDN-based VANET. These metrics are part of the elements that characterize vehicle mobility. The aforementioned issues can be mitigated by using some alternative mechanisms such as the mobility prediction. As explained later, this work proposes two schemes for mobility prediction. A short-term, computationally less complex, deployed at the data plane level and a long-term, relatively more complex, deployed at the control plane level.

Considering the aforementioned characteristics of NDN - the hierarchical content naming scheme, the flexible content retrieval and in-networking caching support - and the fact that most of the communications in VANET follows a content-centric approach [251], NDN have been seen as a highly suitable architectural model for VANET, order to mitigate IP related issues in VANET, such as location-based naming, setup and maintenance of delivery paths and session establishment before data transfer. As such the integration of VANET and NDN, has been proposed by many researchers. The current integration proposals are based on deploying new applications designed from scratch to leverage the characteristics of NDN, redesigned caching schemes, routing and forwarding strategies designed with mobility support [140], by introducing modifications into the core NDN's architecture or proposing new architectural guidelines for adapting NDN for VANET [112]. However, even this integration introduces several issues and challenges that still lacks effective solutions.

Therefore, the motivation behind this thesis is the need for an efficient content delivery model capable of responding to the increased demand of network applications. The proposed solution is meant to close the gaps and challenges presented by the integration of NDN and VANET. Specifically, it pretends to study and propose solutions for forwarding and routing for NDN-based VANET, considering the gaps assessed by the study in [232], and taking in consideration that a good forwarding and routing model should conform with the following functional requirements [200]: a) low latency; b) capability of adapting its addressing mode to unicast, broadcast and multicast; c) high delivery rates; d) at most, cause low additional load; and e) adapt itself to the specific VANET environments (i.e., highway, suburban, urban). Moreover, the thesis should contribute to the NDN community by providing additional experimental results on the role of mobility prediction applied to forwarding and routing for reducing the Transmission Overhead (TO).

1.3 Aim and Research Questions

The advantages of using NDN for developing VANET has been demonstrated by several studies including the ones surveyed in [230]. Section 3 presents the state-of-the-art on improving VANET by NDN and highlights the main existing gaps on the proposed solutions.

This thesis aims to advance the integration of NDN-based VANET in ITS by providing an efficient communication model that closes existing gaps. The gaps were extracted from the state-of-the-art in content dissemination in NDN-based VANET. It focuses the main efforts on routing and forwarding. We further extend our research objective in the form of three research questions, as presented next.

- Broadcast storm is a well known problem that can seriously compromise the performance of the

network. We consider the hypothesis that knowing in advance the trajectories of the nodes and being capable of predicting and selecting a specific node to forward packets to the destination, instead of broadcasting them, could be advantageous for reducing network traffic. Thus, one of the research question we investigate and further answer is *Research Question (RQ)1: whether integrating mobility prediction could reduce network traffic and provide better utilization of the network resources*. This question is investigated in two distinct levels, at the data and control plane using STMP. Then, LTMP is used in both planes only for static nodes, i.e., RSU.

- Taking in consideration one of the functional requirements for a good routing (i.e., one that must be able to adapt itself to the specific VANET environments), as presented by [200], the thesis also investigates *RQ2: to what extent the awareness of the network environment could play a significant role for efficient routing in VANET*. Hence, for routing decisions the proposed model takes in consideration various environment factors such as the network density, the type of content, and the communication models.
- Moreover, considering the importance of the caching system, the thesis investigates *RQ3: to what extent selectively sharing cached content among vehicles for further helping the routing decision might reduce the need for broadcasting and hence reduce the network traffic, when no FIB entry matches the requested content*.

As aforementioned, the proposed RQs will guide the development of the thesis and at the end, the proposed model should conform with the functional requirements for a good routing protocol, as presented by [200].

1.4 Assumptions

The model is designed under the following assumptions:

- Every node in the network is equipped with a navigational system, such as GPS, capable of accessing their mobility information (i.e., node geographical location and speed). This is despite the vulnerable utilization of such systems under the mercy of the sky conditions.
- The nodes are freely willing to share their mobility information and their Content Store (CS) content, despite the security, privacy and trust issues.
- We consider that, in general, storage and processing power are not a restraint aspect in VANET, as referred by [220].

It is important to note that these assumptions simplify certain real-world constraints, particularly regarding security, privacy, trust, and hardware limitations. However, they enable a clearer evaluation of the model's core functionality under idealized conditions.

1.5 Research Methodology

This work follows the Design Science Research Methodology (DSRM) [189, 196] for developing the proposed solution. The DSRM process model includes the following steps:

1. Identification of the problem and the motivation;
2. Definition of the objectives;
3. Design and development of the solution;
4. Demonstration, where a suitable context is selected to test the proposed solution;
5. Evaluation; and
6. Communication, where the results of the proposed solution are presented to the community.

A set of preliminary work has been performed in order to better understand the NDN architecture and associated tools for simulations. This section presents the undertaken work and all steps that took place on developing the present thesis.

NDN is a network layer protocol, originally designed for wired networks. Being a network layer architecture, an initial understanding of the forwarding and routing mechanisms used by the architecture was needed. For this regard, a technical report in [231] was developed. The technical report was used to study the NDN architecture in general, and survey the routing protocols designed from scratch or the ones adapted from the existing ones already working for IP-based networks.

To further understand the NDN' *innards*, another study, and this time a simulation-based technical report, was developed. This work aimed at understanding the caching system in NDN, as well as the simulation environment used for testing this architecture. The aforementioned technical report was, then, updated and adapted for publication as [234].

In order to better define the extent of contributions of this thesis, an SLR was performed in [232] (is presented in Section 3). The aforementioned review concluded that a still scarce number of solutions were proposed for NDN-based routing in VANET. The existing studies do not take advantage of mobility prediction in order to improve forwarding. Furthermore, although storage and computational resources are not restraints in VANET, thus their usage can more freely be explored, the proposed solutions do not leverage the in-network caching for better routing decision. The result of the SLR were used to extract the gaps presented by the state-of-the-art and define the specifics of the proposed model. The SLR focused more on routing, caching and security. The choice on the three topics was motivated by the fact that there was already several similar surveys on forwarding, as justified on the manuscript. Afterwards, a section on forwarding was added to the results of the SLR. The mentioned section is only present in this thesis and not on the published manuscript.

A previous experimental study was performed in [234], and it aimed at accessing the effect of the number of CS in a network. Moreover, it allowed to study in deep the caching mechanisms in this architecture. Although based on a static network, the mentioned study contributed on some of the decisions (e.g., definition of the caching model) taken on designing the present model.

The initial idea of the model developed in this thesis was presented in the 4th Symposium on Electrical and Computer Engineering, hosted by University of Porto (see [19]). Then updated and published in [230]. The idea was based on developing a context-aware forwarding and routing model for NDN-based VANET, enhanced by mobility prediction and caching. Hence, a survey on mobility prediction schemes was performed, and two mobility prediction schemes (based on Extended Kalman Filter (EKF) and HMM) were developed. Although processing capacity is considered as not being constraint for VANET, it is still not obvious at what extend this capacity can really be supported. Even considering a hypothetical above-average processing power, the short-time duration connectivity among vehicular nodes, requires that processing time of these algorithms be as short as possible. Having this fact in mind, a compromise between the accuracy and processing complexity, in choosing a better solution for mobility prediction, is of paramount importance. The work by [298] presents a survey on mobility prediction mechanisms based on Markov Chain (MC), HMM, Artificial Neural Network (ANN), Bayesian Networks (BN), and Data Mining (DM). The work by [2] presents a survey and a taxonomy on prediction-based protocols for TCP/IP-based VANET for routing, forwarding, traffic management, road safety, and general purpose protocols. These studies were used as baseline to review and select a better mechanism for the schemes developed in this thesis.

The model was deigned, implemented and evaluated. The simulation results were statistically processed using MatLab[®] and *bashscript* (for parsing and extracting data from trace files) and SPSS[®] (mainly for the first works in [234, 232]) and later, trace file parsing and statistical presentation using Python libraries.

1.6 Thesis Contributions

The present work investigates and proposes mechanisms to increase network performance for NDN-based VANET. The main contributions produced by the thesis are summarized as follows:

- A performance evaluation of some of the known caching policies deployed in NDN Simulator (ndnSIM). Conclusions from this study provided a correction to a previously published work in terms of the impact of the number of CS in the network to its global caching performance.
- A comprehensive literature review on caching, forwarding, security and privacy, and routing for NDN-based VANET.
- A forwarding strategy based on mobility prediction and using the overheard packets instead of resorting to a beaconing process. The proposed model aimed at reducing the network traffic

without additional packet purposefully inserted in the network for tracking vehicles movements. We further investigate and compare the model with its beacon-based mode. In the beacon-based mode a mechanism relying on a Bloom Filter (BF) [45] for content sharing is provided. For new content discovery, the RSU sends a periodic request for which the producer responds with a special Data packet, announcing its new content and attaching, by means of a BF, the contents stored in its CS.

- A routing model based on mobility prediction and overheard packets instead of resorting to a beaconing process, for tracking and updating the FIB.
- Two algorithms for predicting the mobility of nodes are proposed.
- Overall, we addressed the proposed research questions and we presented their responses on a) how the mobility prediction could reduce the TO and boost network performance, b) how the awareness of the network environment could play an important role on network performance and, c) whether sharing cached content could, also, be leveraged in order to reduce the need for broadcasting beacon when no corresponding FIB entry could be found.
- We provide and discuss the results of extensive simulations performed to assess the effectiveness and performance of the proposed model.

The following references were published throughout the development of this work, in order to address the research questions presented in the previous section. For each work we present the current number of citations, extracted from google scholar.

- NDN Content Store and Caching Policies: Performance Evaluation (In: Computers, 11(3), 37) [234] - although not focused on the vehicular network, this work aimed at studying and evaluating the caching replacement policies usually deployed in NDN networks. The obtained results and the results from [144] for caching placement strategies, are used to choose a better caching scheme for the present thesis. With 22 citations (Scopus), 23 citations (Google Scholar).
- Design of a Context-Aware Routing System Model with Mobility Prediction for NDN-based VANET (In: DCE21 13 Symposium on Electrical and Computer Engineering: Book of Abstracts, 21) [19] - the proposal developed by this thesis was firstly designed and validated by this work, which was presented in a doctoral congress, in Porto. With 1 citation (Google Scholar).
- Context-Aware Routing and Forwarding Model for NDN-Based VANET (In: Lecture Notes of the Institute for Computer Sciences Social Informatics and Telecommunications Engineering Lnicst, 428 LNICST, pp. 18–32) [230] - the proposal presented in [19] was refined and enriched in this this work, then it was presented in an international conference. With 4 citations (Scopus), 6 citations (Google Scholar).

- On the realization of VANET using Named Data Networking: a systematic review (In: International Journal of Communication Systems, 35(18), e5348) [232] - this work aimed at accessing the state of the art on leveraging the synergies from VANET and NDN. From the results of this study a better definition of the focus of the thesis was produced. With 10 citations (Scopus), 14 citations (Google Scholar).
- Context and Mobility-Aware Forwarding Model for V-NDN (CMAF): Context and Mobility-Aware Forwarding Model for V-NDN (In: Electronics Switzerland, 13(12), 2394) [233] - this work aimed at partially presenting the forwarding strategy. 4 citations (Scopus), 4 citations (Google Scholar).
- Mobility-Aware Routing Model for V-NDN (MARV) - Ongoing work preparation for publication.

The model is implemented as described in Chapter 5. To ensure reproducibility, the source code of the developed model is publicly available at https://github.com/6Nokturnal6/ndn-based_VANET.

1.7 Thesis Outline

The remaining of the thesis is organized as follows:

- Chapter 2: **Concepts Background** - this chapter presents the theoretical background that includes the overview of the VANET, the overview of the NDN, the mobility models and mobility prediction theory in VANET.
- Chapter 3: **NDN-based VANET - State-of-the-art** - presents a comprehensive literature review, which was performed by means of a SLR methodology.
- Chapter 4: **Design Overview** - presents the design of the proposed routing and forwarding model. The main sections in this chapter are: Section 4.1 which is an introductory context. Section 4.2 that positions the proposed model among the state-of-the-art, in terms of its methodology for discovering neighbors and content dissemination. Section 4.3 presents the design of the forwarding strategy, which includes the presentation of its novelty. Section 4.4 presents the design of the routing model. The STMP and the LTMP are presented in Section 4.5 and Section 4.6, respectively. And, finally, the conclusions of the chapter are presented in Section 4.7.
- Chapter 5 **Implementation of the proposed model** - this chapter presents the implementation of the proposed model. The main sections in this chapter are: Section 5.1, which justify and presents the chosen network simulator for implementing and evaluating the proposed model. Section 5.2 justifies and presents the chosen road traffic generator besides the procedures used to configure the two road network scenarios used in simulations. Section 5.3 presents the hardware and software specification of the simulation environment. Section 5.4 presents the main modifications applied to the simulator in order to adapt it to the proposed model. The main challenges

encountered on implementing the model are presented in Section 5.5. Finally, the summary is presented in Section 5.6.

- Chapter 6: **NDN-based VANET: Performance Evaluation** - presents the evaluation of the proposed model. The selected metrics are presented in Section 6.1, and Section 6.2 presents the simulation environment. Section 6.3 and Section 6.4 present the simulation results for the proposed CMAF and Mobility-Aware Routing Model for V-NDN (MARV), respectively. The chapter ends presenting the general discussion in Section 6.5.
- Chapter 7: **Conclusions and Future Work** - presents the conclusions, the main limitations of the work in Section 7.3, and the future work in Section 7.4.

Together, these chapters provide a structured progression from theoretical foundations and related work to the design, implementation, and evaluation of the proposed model, culminating in the conclusions and directions for future research.

Theoretical Background

The aims of this chapter is to present the background of the technologies on which the thesis is grounded, given that several theoretical aspects should be mastered in order to better understand what the thesis proposes.

The chapter is outlined as follows: first, the chapter presents, in Section 2.1, an overview of VANET including its architecture, proprieties, applications, security and privacy, and finally its main challenges and research issues. Then, it presents an overview of ICN, including the presentation of NDN architecture, the routing and forwarding aspects in NDN, and finally the caching and security. The synergies from VANET and NDN are, then, presented in Section 2.3. The chapter is finalized by presenting the theories behind the mobility prediction algorithms, in Section 2.4 and the summary in Section 2.5

2.1 Overview of Vehicular Ad hoc Network (VANET)

VANET is a specific area of networking that have been under intensive research, both from academia and industry, due to its foreseen importance to the development of ITS. In fact, the main purpose of VANET is to provide ubiquitous connectivity to mobile users, and efficient V2V communications in order to enable ITS [177]. VANET is a particular case of MANET and presents specific characteristics that make it differently challenging compared to the latter, as presented in the next sections.

2.1.1 Architecture

Communications in VANET are based on the following three network scenarios:

1. V2V or pure ad hoc - which refers to the communication only between mobile nodes, in an opportunistic or ad hoc manner. This scenario is essentially important in environments where fixed infrastructures are inexistent or where communication can not allow the delays produced by routing through the infrastructure;

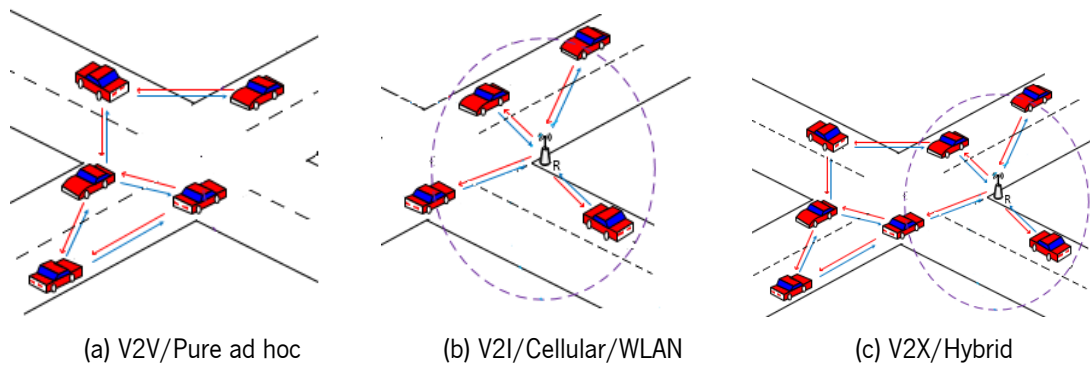


Figure 1: The VANET architectures

2. Vehicle to Infrastructure (V2I) or cellular/WLAN - which is created between mobile vehicles and the static infrastructure (e.g., RSU or Base Station (BS)) in the network. The static infrastructure is mainly used for routing or as cellular gateways and WLAN Access Point (AP) to provide Internet; and
3. Vehicle to Everything (V2X) - the hybrid model where communication can occur from the mobile vehicles to every capable node in the network, and vice-versa [177, 120].

Figure 1 shows the aforementioned three VANET architecture scenarios.

2.1.2 VANET proprieties

The main properties of VANET are [198]:

1. Ad hoc network - an ephemeral network with no previously defined structure;
2. Predictable mobility - the RSU/BS are static. The vehicle trajectory is constrained by the topology and the layout of the roads;
3. Highly dynamic network topology - although the mobility is predictable, the varying directions taken by the vehicles result in a short life duration of links among nodes (i.e., the nodes present intermittent connectivity). This characteristic is predominant on highways where vehicle speed can go up to 200km/h [219];
4. Large scale network - VANET could cover a very large area, from a neighborhood to an entire city;
5. Ability to provide continuous power and high computational capability - as we assumed and based on [220], the power and computational ability of the nodes (vehicles and RSU) in VANET is not a constraint. The nodes are equipped with a variety of computational resources such as sensors, storage, GPS, and processors [220];

6. Partitioned network - the network traffic varies depending on the considered area: rural, suburban or urban. As one move from the rural to the urban area the network traffic tend to be higher, but the traffic speed of the vehicles will be moderate compared to the rural area.

2.1.3 VANET applications

The classification of VANET applications can be done taking into account several factors. Besides the classification into delay-tolerant and delay-sensitive, related to their sensibility with time/delay, requirements such as [83], a) the strategic requirements (e.g., required minimum penetration threshold for the considered application, and its assigned use cases able to provide a usable, valuable customer' service.); b) economical requirements (e.g., perceived customer value of a service which will enable the estimation of the service sales price); c) system capabilities requirements (e.g., the radio communication single-hop coverage, ad-hoc networking capabilities); d) system performances requirements (e.g., vehicle communication performances - maximum latency time, maximum round trip delays); e) organizational requirements (e.g., availability of a suitable organization providing support to security protection); f) legal requirements (e.g., respect of customer's life privacy); and g) standardization and certification requirements (e.g., system standardization), can lead to other class grouping.

VANET applications are quite vast that various authors end up classifying them differently. Although classification naming could be different, application grouping into classes is somehow similar. Typically, applications are categorized into the following classes, 1) *safety*; 2) *transport efficiency*; and 3) *information/entertainment* [105]. Actually, looking at a higher level of grouping, these applications end up being divided only into two groups, *safety* and *non-safety* applications, then a more detailed classification follows into each of the two groups, like the case of classification by [16], for instance. Several authors use the more detailed scheme, like the aforementioned and referred by [105].

The study by [68] categorizes applications into the following classes, 1) *safety* - which aims at avoiding and decreasing the number of road accidents; 2) *Efficiency* - which aims at improving the vehicle mobility within the public roads, 3) *Comfort* - which aims at providing information from vehicular services to drivers making their trip more comfortable and enjoyable; 4) *Interactive entertainment* - aiming at distributing and delivering entertainment-related information to drivers and passengers; 5) *Urban sensing* - aiming at producing or collecting urban monitoring information, and share data of common interest.

The European Telecommunications Standards Institute (ETSI) [83], has identified a Basic Set of Applications (BSA), classified and grouped them into the following classes, 1) *Cooperative road safety* - to improve the road safety; 2) *Cooperative traffic efficiency* - to improve traffic fluidity; 3) *Cooperative local services*; and 4) *Global internet services*. Both classes (3) and (4), aiming at advertising and providing on-demand information to passing vehicles on either a commercial or non-commercial basis, including Infotainment, and comfort.

As referred before, whilst the nomenclature may be different, the grouping itself ends up being quite

similar. We consider the classification from [68] more representative, and meeting our idea. Thus, we borrow and adapt the class nomenclature, from that work. Based on classification and requirements presented by the aforementioned studies, we present in Figure 2, the final classification and the characterization adopted for this thesis.

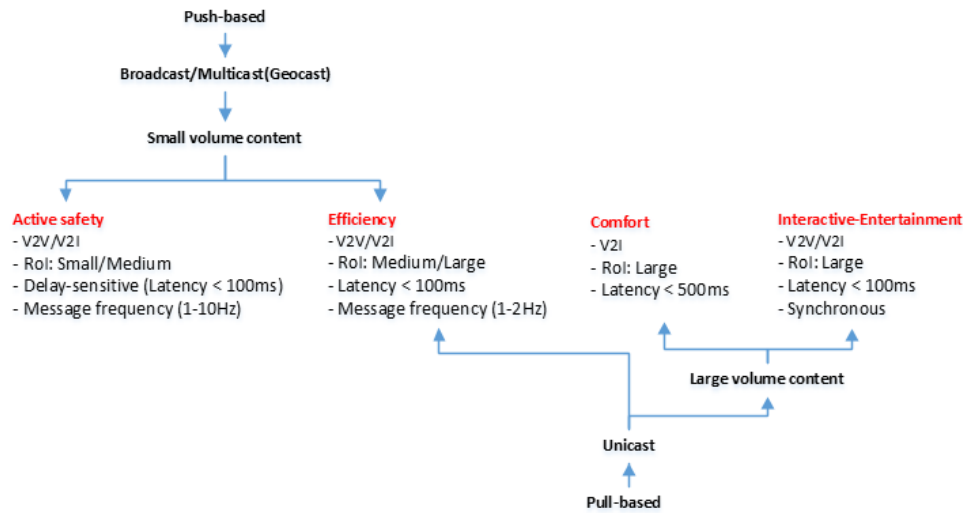


Figure 2: Network-based classification of VANET applications

In order to give examples of some applications for each class, we resort to the applications requirement descriptions from [83].

As an example of active safety application, one can consider the stationary vehicle warning, where a vehicle being dangerously immobilized on the road (caused by an accident, or any other reason) must alert other approaching vehicles of the risk for them, associated to this dangerous situation. Vehicles should be able to broadcast periodic V2X notification messages, informing the neighboring vehicles about its current immobilization situation. These applications have a Region of Interest (RoI) - the area where the broadcast information makes sense - relatively small or medium. Applications grouped into this class are delay-sensitive, allowing maximum latency of 100ms [83].

For applications in efficiency class, one can consider a situation where a vehicle informs the approaching vehicles about some abnormal traffic conditions and eventually issue some recommendations in case of traffic jam. Applications into this class should be capable of broadcasting periodic traffic information and also be able to respond unicast requests from other vehicles. This class also includes delay-sensitive applications but relatively more tolerant than the active safety class. The active safety and efficiency classes are based on communication of periodic, short-lived and small volume messages appropriate for broadcast.

For applications in comfort class, one can consider the media downloading applications, where a relatively large volume file (e.g, video or audio files) should be downloaded. Given the large volume file, the high mobility of vehicles, and the fact that these are delay-tolerant applications, the V2I communication

model - where the infrastructure is static - can be leveraged to benefit applications in this class. For applications in interactive-entertainment class, an example can be the video games, or video conference streams. Applications in this class are generally delay-sensitive, and composed by large volume content. Note that some applications in this class may be multicast, but to keep the simplicity here, we consider all as unicast. Applications into Comfort and interactive-entertainment classes are pull-based.

2.1.4 Security, Privacy and Trust

Different active safety and efficiency-related information is constantly shared by nodes in VANET. Drivers make critical decisions based on this information, then its correctness must be assured. Key aspects in VANET communication are security, authenticity, and integrity [208]. Common VANET privacy and security vulnerabilities include [210, 92, 216]: a) jamming - a node can deliberately send interfering messages that can prevent communication among vehicles in their communication range; b) forgery - where a node can forge and disseminate false emergency information; c) traffic tampering - the attacker node can modify traffic information eluding the drivers about the real state of the traffic; d) impersonation - a node can masquerade and act as an emergency service and disseminate false emergency information; e) location tracking and privacy violation - achieved from the collection of various information about the mobility and preferences of a given driver; f) sensor data faking - achieved by bypassing the sensor data and fake information such as the speed or position; g) Sybil attack - where a node use multiple identities to send multiple messages as if it was multiple sources sending the messages; and h) black hole attacks - the dishonest node pretends to have the shortest path to content sources, eluding the other nodes to choose this path but the packets never reach the destination.

Several security mechanisms have been proposed in order to protect privacy, integrity, achieve anonymity and authenticity, which are also key aspects of VANET communication [208]. Some of these schemes include [41, 296, 167]: a) trust-based mechanisms (which can be grouped into authentication-based and reputation-based [216]); b) group communication; c) data fusion schemes; d) identity-based signature schemes; e) pseudonym approaches; f) Public Key Infrastructure (PKI); g) k-anonymity schemes; i) aggregate signature schemes; and j) certificate revocation.

2.1.5 Main challenges and research issues

The main technical challenges which compose the main areas for research in VANET [132, 140, 105, 242] are: a) Signal fading; b) Bandwidth limitations; c) Short period of connectivity; d) Small effective diameter; e) Security and privacy; f) Routing protocols; g) Data administration and storage; and h) Time constraints.

Signal fading (related to the propagation model) - in a road vehicular communications where buildings and other vehicles are in between two communications nodes, the transmitted signal tend to deteriorate.

With more obstacles as the case of urban areas, the signal fading may lead to infeasibility of the communication. Although rural areas with less buildings may benefit the signal propagation, the high speed of the vehicles will affect the signal, making it difficult to model the channel.

Bandwidth limitations - VANET applications use a limited range of bandwidth frequency, as such in a highly dense environment, inefficient use of the bandwidth may result in a channel congestion.

Short period of connectivity (link intermittency) - due to high mobility of nodes which result in a highly dynamic topology the connectivity between nodes in VANET is inevitably short, causing the instability of the routing paths. Exchanging information in a such small period of time demands the development of more advanced and rich network topology model, different from traditional models that have larger period of interaction between sender and receiver.

Small effective diameter - this leads to the challenges in maintaining the global network topology and resulting in problems when applying existent network routing protocols to VANET.

Security and privacy - being an open network where any node is allowed to join, these challenges demand a good balance. For trustworthiness a receiver would require to know the source of the received information. However, this aspect could violate the privacy of the sender node. Having in mind that one of the applicability of VANET is the safety of vehicles and pedestrians (it directly involves human lives), the security and trustworthiness of the information must be guaranteed.

Routing protocols - are one of the fundamental elements in networks and in special for VANET. Routing protocols are responsible for initiating and managing routes to facilitate multi-hop communications. Routing is a critical challenge and the difficult in resolving this challenge comes from two of the main characteristics of VANET: the high mobility of nodes and the highly dynamic topology.

Data administration and storage - for large scale networks, the number of nodes can increase and generate a large amount of data to manage.

Time constraints - applications in VANET can be divided into two groups concerning their sensibility with time: a) delay-tolerant - mostly the entertainment applications; and b) delay-sensitive - which is related to the active safety applications and where reliability, responsiveness, and data quality is critical. Time constraints must be assured in order to enable the driver take right and timely decisions, when required.

2.2 Overview of Information Centric Networking (ICN)

The actual IP-based Internet is already a half-century aged, host-centric network. In this network, communication between hosts demands a previous knowledge of their respective addresses in the network. The protocol was originally designed for wired host-to-host communications. Since its standardization up to now, network communications have changed profoundly, from the host-centric to content-centric model. Indeed, content consumers are now more interested on the content itself rather than on its physical location.

The rapid growth of technologies, including electronic commerce (e.g., Amazon), live multimedia (e.g.,

Netflix), social networks (e.g., Facebook, WhatsApp), leads to an increasingly accentuated inadequacy of the IP-based Internet, demanding the development of new, and more adequate information-centric solutions. In fact, the need to better cope with the network disconnections and disruptions, and address other IP-based network problems including host mobility, network security, data privacy, network scalability and multi-path routing, which remain challenging for this network, motivated the development of the future internet architectures based on NDO as a core architectural principle. The approach of these architectures is commonly called ICN [9]. Several research projects based on this concept emerged [85]. Some of them include the CCN [118], Network Information (NetInf) [70], Data-Oriented Network Architecture (DONA) [148], Publish-Subscribe Internet Routing Paradigm (PSIRP) [247] and its continuation project Publish-Subscribe Internet Technologies (PURSUIT) [87]. Where NDN [302, 301], a particular implementation of CCN, has been seen as the most promising project [252, 301]. The aforementioned architectures share the same principles (e.g., content-centric, self-identifying NDO for supporting pervasive and ubiquitous caching) but differs in specific aspects including dissemination, naming and security schemes [239].

Table 2 summarizes the main characteristics that distinguish the architecture from each other. Some specific differences are in terms of

1. The ICN Application Programming Interface (API), that is defined as the way of which the NDO are requested and delivered [9] - whilst PSIRP is based on a publish/subscribe-like approach, in which the client registers a subscription and is notified when content is available, the other architectures use a simple synchronous get scheme where a consumer asks for a desired content, whenever it needs the content;
2. Namespace - where differently from the other architectures that use flat schemes with structure, NDN uses a hierarchical scheme (which is semantically meaningful, making it expressive and application-specific [270]), providing aggregation and better scalability;
3. NDO granularity - where, instead of objects, NDN adopts the packet level granularity, favoring better caching management;
4. Routing of NDO request - where NDN adopts name-based routing and does not require external services such as Resolution Handler (RH) or Named Resolution Service (NRS), for name resolution; and
5. Caching - whilst in NDN any node is able to cache content, the other mentioned architectures limit caching to a specific subset of nodes, and the storage is managed by the RH in DONA, the Rendezvous system in PSIRP and the DHT-based resolution system for NetInf [239].

Differently from the ICN approaches mentioned early in this section, the Recursive InterNetwork Architecture (RINA) [71, 94] is proposed considering that current TCP/IP protocol stack presents many flaws, including 1) the fixed number of layers; 2) existence of multiple protocols per layer (each use case generally requires new protocol); 3) little commonality among layer, that is, protocol layers are independently

Table 2: Summary of characteristics of ICN approaches [9]

	DONA	CCN	PSIRP	NetInf
Namespace	Flat with structure	Hierarchical	Flat with structure	Flat with structure
Name-data integrity	Signature, PKI independent	Signature, external trust source	Signature, PKI independent	Signature or content hash, PKI independent
Human-readable	No	Possible	No	No
Information abstraction model	No	No	No	Yes
NDO granularity	Objects	Packets	Objects	Objects
Routing aggregation	Publisher/explicit	Publisher	Scope/explicit	Publisher
Routing of NDO request	Name-based (via RHs)	Name-based	NRS rendezvous	Hybrid NRS and name-based
Routing of NDO	Reverse request path or IP direct connection	Reverse request path using router state	Source routing using BF	Reverse request path or direct IP connection
API	Synchronous get	Synchronous get	Publish/Subscribe	Synchronous get
Transport	IP	Many including IP	IP/PSIRP	Many including IP

designed from each other; 4) security is provided as extensions rather than by design; 5) mobility issues given the need to create a session for host-to-host communications.

RINA is designed based on the principle that networking is a distributed Inter-Process Communication (IPC), where application processes communicate via an IPC facility (IPC layer) [71, 117]. RINA does not present a fixed number of layers. Instead, it recursively replicates the IPC layer, i.e., a Distributed IPC Facility (DIF), over different scopes. Where each DIF consists of a set of communicating processes, and performs the same functions at different scopes [94]. The use of self-consistent Data in the CCN approaches allows in-network caching that facilitates content retrieval and reduction of traffic congestion. However, RINA is not designed with this characteristic.

Additional information of former and current CCN projects including their standardization are comprehensively detailed by [9, 276, 31, 289].

2.2.1 Architecture of NDN (a specific realization of ICN)

NDN is a content-centric and name-based architecture. The contents identification in NDN is based on names instead of IP addresses. Identifying contents through names provides their dissociation from their location, application, storage, and means of transportation [151]. This propriety results in the advantage of being able to ubiquitously store and share content among several consumers, benefiting the mobility and other communication requirements of VANET applications and services. Furthermore, communications in NDN are session-less, meaning that there is no need for a session-based communication and continuous channel availability as is the case of TCP/IP-based networks. This is one more characteristic that combined with the use of NDN in-network caching leads to a time decoupling design where the content producer is not required to be connected during the content transfer to the consumer [140].

The following architectural principles guided the design of NDN [302]:

1. The use of hourglass architecture, focusing minimum functionality in the network layer, similar to the TCP/IP protocol stack. The network layer in NDN should support scalability, security, resilience,

and efficiency (to support multipath routing and management of packets in content stores). Two additional layers (security and strategy) are appended to NDN protocol stack. The new layers provide: a) security layer - security for packets in the network; b) strategy layer - strategically supports the forwarding plane in managing the statuses of each connection, for a better forwarding decision;

2. Security built into the structure - security in the NDN is naturally guaranteed by forcing the producer to sign all Data packets;
3. NDN traffic is self-regulating - the forwarding plane has the responsibility of recording the state of each node. This information can later be used to make decisions about traffic behavior;
4. Separated routing (control) and forwarding (data) planes - this way, each plane can be developed and improved with less interference from each other [300].

NDN architecture implements the forwarding strategy per node. The responsibility of the data plane is to make decisions including: a) which packets to forward, and which interfaces to use; b) which level of unsatisfied packets should be allowed; c) what are the relative priority of different packets. Each node also makes local decisions to balance traffic by forwarding through multiple and alternative paths [300].

NDN adopts pull-based communication model. For communication, two types of packets are adopted: the *Interest* and the *Data* packet. See Figure 3. A consumer initiates the communication process by sending a request by means of an Interest, for the desired content.

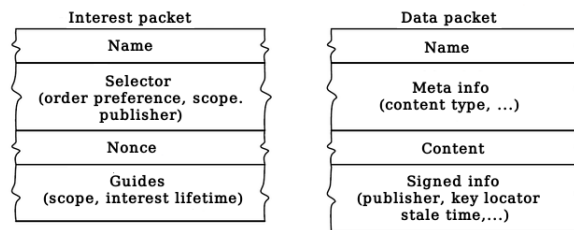


Figure 3: NDN packets. Interest (left) and Data (right). Details for packets specification in [202]

When an Interest is sent out, it is forwarded (upstream path) through different nodes towards the content provider. When corresponding content is found, it is sent back (downstream path) to the requester node following the reverse path used by the Interest (i.e., by breadcrumbs). The status information created and saved along the upstream path is used to guide the Data back. With this approach, Data looping is avoided. Interest looping is avoided by combining the Content Name (CN) with a *Nonce* - a random number generated to unambiguously identify an Interest. Figure 6 shows the forwarding process within the NDN Forwarder Daemon (NFD) [6].

Each NDN node contains a structure composed by the CS, the PIT and the FIB, for packet (Interest and Data) management.

The CS is leveraged to increase the probability of content sharing, saving bandwidth and decreasing the response time for content request. NDN nodes temporarily store (cache) Data forwarded through them. The cached copy will be forwarded to any consumer who requests it, without the need to request it again from the content source. The CS is similar to buffers on IP-based routers with a difference in size and the content replacement policies. Since IP-based communications are host-to-host, the cached packet is of no further use as soon as it is forwarded to its destination. In NDN, as long as the packet is still valid it can be shared with any node.

PIT is used to store information on pending and on recently satisfied Interest. If an Interest reaches a given router and misses its corresponding content, it is further forwarded to other nodes. However, before it is forwarded, its CN and ingress Face are registered on PIT. This information is then used by the breadcrumbs procedure. On reception of several copies of the same Interest, PIT performs their aggregation to prevent the forwarding of multiple requests for the same Interest. The first Interest is forwarded and for the following only the ingress Face is registered, so that later a copy of the content can be served through this Face. The Interest life in PIT is generally equal to Round Trip Time (RTT). Anyway, the pending Interest can be removed from PIT even before RTT elapses. In this case, the request will not be satisfied and it is the responsibility of the consumer to request it again. PIT is one of the key elements for efficiency of NDN. The state of this table in each node has crucial information. It includes the set of ingress Faces, providing natural support for multicast. By controlling its own size, PIT is able to control the rate of Data arrival. Combining PIT with CS, NDN removes the dependency of the transport layer for congestion avoidance. In addition, the status of PIT can be exploited to mitigate DDoS attacks [302].

FIB stores the routing information and can be populated manually or by a name-prefix based routing protocol. NDN FIB is similar to the routing table on IP-based networks, with the difference that it contains name prefixes instead of IP addresses. The incoming Interest that misses the respective contents in a specific router cache, after having been added to PIT, is forwarded to next-hop based on the information provided partially by FIB. The existence of more than one outgoing Face registered on a specific FIB entry for a given Interest forwarded, indicates that this Interest was forwarded through more than one path. If none of the FIB entries matches the Interest, then the Interest is forwarded through all node's Faces.

For wired networks, in case of failure forwarding an Interest due to path congestion, insufficient information in FIB about the prefix, or any other error, a Negative Acknowledgment (NACK) is sent back downstream, with information about the possible cause of the failure.

2.2.2 Routing and Forwarding in NDN

The core characteristic of NDN paradigm shift compared to the host based, is the content naming and its subsequent name-based routing and forwarding. In NDN, all contents are unambiguously identified and retrieved by their names - a CN. The CN is opaque to the network, which means that NDN routers know nothing about their meaning, however, they know how to delimit each component of it. This scheme brings the possibility for applications to be able to choose the naming standard and allows naming schemes to

grow individually within the network [119].

The CN is unique. However, the content it identifies may be spread over several locations on the network, from servers that share content to different routers across the network. It must be globally unique if used to access content globally, but does not need to be global for local networks where it can be based on the local context. Differently from other similar projects that adopt the flat scheme, NDN adopts a hierarchical structure for the CN, allowing the routing scalability. The CN lookup in the CS or PIT is performed by searching with one-to-one correspondence. In FIB, the lookup is performed by Longest Prefix Match (LPM) [302].

NDN is designed with routing and forwarding planes separated, as illustrated in Figure 6. This separation allows their development separately. With a complementary role to the forwarding, NDN uses routing to generate and disseminate the initial topology, its policies, and to manage the long-term changes [288]. It populates FIB and in coordination with the forwarding plane it makes the assessment and testing of the forwarding interfaces. Whereas the routing plane calculates and decides globally on routes availability, the forwarding plane makes per node decisions about the preference and use of these routes based on their performance and status. That is, routing plane finds and calculates costs of routes towards the requested contents and provides them to the forwarding plane.

The forwarding procedure can be summarized as the following (see Figure 4): When an Interest is received, its corresponding Data packet is first searched in the CS. If not found, the registration of the packet is searched in the PIT. If not found then it is registered and forwarded to the FIB. If it is found, the correspondent registration is added and the new packet is discarded. If the Interest is not registered with the FIB, a NACK (Only defined for P2P communications) is returned downstream, if it has been registered, then is forwarded to next-hop. Data packets follows the breadcrumbs path created by the Interest forwarding process.

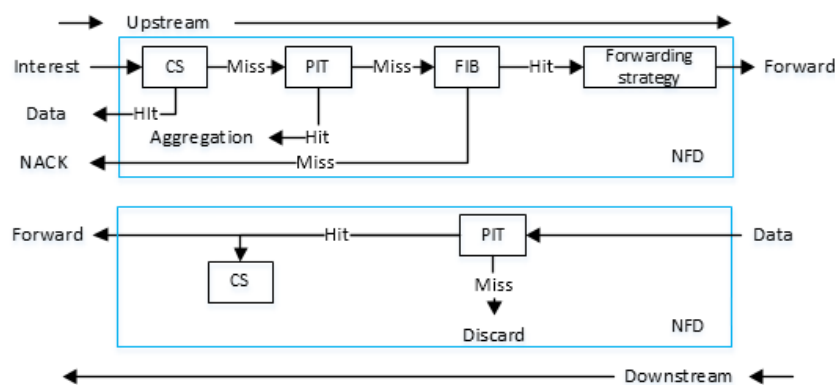


Figure 4: The forwarding procedure in NDN architecture.

Figure 5 shows a hypothetical routing procedure in NDN. As referred, routing is name-based. FIB in each node stores the name-prefix and corresponding next-hop(s), in terms of the Face to which the Interest should be sent towards the content provider. There can be multiple stored next-hop for each name-prefix. Moreover, due to node mobility, stored next-hop(s) may be obsolete and thus, not leading

to any content. Content may be fetched from different sources (which can have the content stored on their CS), or through different paths to the same content provider (multipath routing). For instance any hypothetical file *report1*, a content under */uminho/algorithm/docs*, required by node *A*, can be fetched from path $[R1, B]$ or $[R1, R2, B]$, where *R1* and *R2* are intermediate routers.

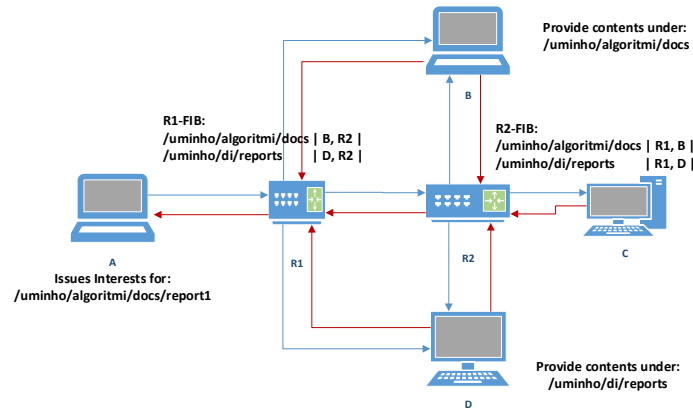


Figure 5: A simple representation of two FIB's entries for each routers, R1 and R2.

The forwarding plane forwards Interest and contents to the next-hop based on these routes or based on broadcasting. Instead of using routing protocols, small NDN-based environments can learn paths and discover the availability of Data by flooding and self-learning [226].

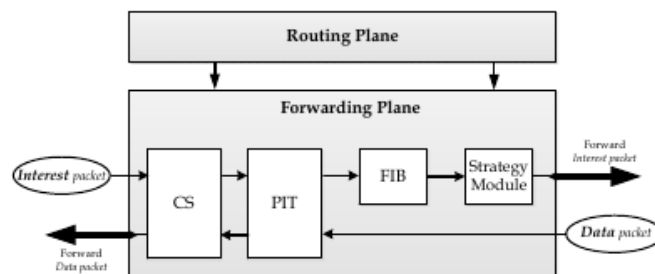


Figure 6: Routing and forwarding planes in a NDN router [91]. The routing plane is responsible of computing and keeping updated the routes (in FIB) to the content providers, whereas the forwarding plane makes the assessment and testing of the forwarding interfaces.

In contrast with the IP-based network where the forwarding plane is stateless and depends strictly on the routing plane, in NDN the forwarding plane is similarly intelligent and adaptive, and able to make forwarding decisions based on the interface status in each node.

Although NDN can work without a routing protocol given the intelligence of the forwarding plane, which can detect and recover by itself from any situation of network failure, the need for routing is exhaustively investigated and justified in [288]. The aforementioned study raised three questions (i.e., whether the forwarding plan would be capable of retrieving Data on a stable network; if the plane would be capable of efficiently handling link failures; and if it could efficiently handle link recovery) in order to study whether routing would still be as essential as it is in IP-based networks. The aforementioned study concludes that

although NDN forwarding plane is smart and capable of handling data dissemination based on flooding and self learning, routing is still necessary. In fact, even for a highly dynamic network, proposals aiming at maintaining and managing some routes in FIB have been proposed, see the state-of-the-art in Chapter 3.

Routing in VANET is challenging (if not infeasible at all [97, 239]), due to the characteristic (e.g., high node mobility and intermittent connectivity) of these networks. For this reason, the majority of research for data dissemination in NDN-based VANET resort to broadcast for data dissemination, and do not keep and manage routes in FIB (i.e., are based on forwarding strategies and not on routing). Nonetheless, there are several proposed solutions designed to populate and manage routes in FIB and use them to disseminate data. These routes would be more cluster-based than global due to the mobility of the nodes.

2.2.3 NDN In-network Caching

One of the most important properties of NDN is the capability of caching contents. This is only possible because NDN content is a self-consistent unit [46]. Caching is performed at the node level. Each node is capable of caching content. Resource availability is the only limitation for caching. A cached content can be forwarded to any node requesting it, without the need to retrieve it from the original content source. This property provides the possibility of sharing (speeding up content retrieval and delivery), saving bandwidth and decreasing the response time for content request.

NDN is capable of communicating asynchronously, that is, no simultaneous presence of consumer and provider is required. This capability is further empowered by the in-network caching, which also provides the store-and-forward capability required under intermittent connectivity conditions [23].

2.2.4 NDN Built-in Security

IP-based VANET inherit the security issues and challenges from that underlying technology, in which the security is based on creating a secure host-to-host channel, to secure the content transmission between them, for each session. In this mechanism, the secure information cannot be re-used when a session expires [140]. To worsen this aspect, the main VANET properties (i.e., the varying density, high node mobility, and the time-critical nature of the most VANET applications) enforce that the security mechanisms be time- and bandwidth-efficient [216].

Security, authenticity, and integrity are key aspects in VANET, and NDN inherently provides them [208, 46]. Instead of securing the communication channel, NDN security mechanisms are provided to the content itself [140]. Each NDN Data packet contains a signature piggyback with content source information. Different Encryption/Decryption techniques are used to provide access control [46].

2.3 VANET enhanced by NDN

Considering the aforementioned NDN characteristics - the name based content identification for flexible content retrieval, built-in security, and in-network caching combined with the fact that most applications in VANET are content-centric [251], NDN have been seen as a highly suitable architectural model for VANET, in order to eliminate the TCP/IP related issues. As such, the integration of both VANET and NDN has been proposed by many researchers. Current integration proposals are based on optimized forwarding strategies with customized naming and caching schemes, clean application design, and seamless mobility support, by introducing modifications into the core of NDN, or by proposing new architectural guidelines for adapting NDN for VANET [140]. However, even this integration introduces several issues and challenges which are still lacking effective solutions.

2.4 Mobility Prediction in VANET

The key characteristics of VANET (i.e., highly dynamic topology, and intermittency of connectivity) make it difficult to build and maintain FIB [97, 239]. Although these characteristics impose specific challenges for several applications, they can be exploited, for instance, for developing better movement management. In fact, due to movement constraints imposed by road topology and traffic conditions combined with the repetitive characteristic of users' movements, the future movements of the vehicles are usually predictable. As concluded by [155] on the predictability limits of vehicular mobility, there exist some strong regularity in the daily vehicular mobility in both temporal and spatial dimensions. This regularity can be exploited to predict the vehicular mobility with high degree of prediction accuracy.

Some representative applications that take advantage of prediction algorithms in VANET include routing, forwarding, traffic management, road safety, handover management, resource management, and location-based applications [2, 298]. Based on their objectives, the prediction algorithms can be categorized into 1) link stability prediction; 2) location prediction; 3) trajectory prediction; 4) travelling time prediction; and 5) collision prediction [2]. Specifically for routing, the aim is to predict the next positions (location prediction), the speed, and moving directions of the vehicles in order to increase the overall performance of the network by reducing the need for frequent broadcast of beacons, which are used to discover the nodes location (i.e., neighbors).

The trajectory prediction aims at predicting a set of intermediate location that the vehicle will follow toward its destination. The set of intermediate locations is mostly used for forwarding. Having an early knowledge of future trajectories allows the selection of a better relay vehicle, which provides the best forwarding performance. As we are interested in having several future locations, the scheme should perform a relatively long-term prediction. The location prediction, in turn, is commonly used to improve position-based routing protocols [2]. The location prediction, which performs a generally short-term prediction, provides a good opportunity to get and update the information about the neighbor nodes, avoiding a very frequent broadcast of beacons.

The maintenance of routes in VANET can be relatively achieved by increasing the frequency of beacon broadcast. This solution, however, brings in negative results. In a dense environment such as the urban scenario, frequent beacon broadcasting can result in traffic congestion, packet collision, increase delivery delays and, consequently a deteriorated QoS and Quality of Experience (QoE).

A key factor to have in mind when choosing or proposing a mobility prediction scheme for VANET is tied with the main characteristics of this network. Although computational processing power is not a constrain in VANET, the real time characteristic of some of its applications and its short-links duration makes it necessary to develop or choose a mobility prediction scheme with relatively low processing complexity, in order to be able to offer better QoS and QoE for vehicular users.

Mobility prediction is a component of mobility management [75] and can be viewed as the means in which a system proactively estimates the future location of mobile nodes. Mobility prediction algorithms can be categorized into various, and different groups depending on the domain that is used as reference. For instance, authors in [62] use the following categorization: 1) domain-independent algorithms - which are based on past node mobility history to extract context and predict the future location, usually requiring high computational capacity for algorithm training; and 2) domain-specific algorithms - which use the geometry of node motion and the semantics of the symbols in node's mobility history. The work by [104] considers: 1) deterministic model, which uses the vehicle displacement/kinetics to compute the future position; 2) history-based, where the model learns from past historic movement patterns to predict future position; and 3) stochastic models, where the focus is on correcting the prediction error using probabilities. With only different naming, the categories domain-independent, and domain-specific algorithms from [62], are respectively equivalent to history-based and deterministic models, from [104].

The use of mobility prediction models range from cellular, mobile networks and vehicular networks. Generally, mobility prediction in mobile network benefits location-based applications and provides efficient resource and handover management, which can avoid high degradation of QoS or QoE [298].

State-of-the-art location and mobility prediction approaches include but, not limited to: a) Dead Reckoning (DR); b) MC; c) HMM; d) ANN; e) Data mining; and f) Filtering.

DR - uses last known position, displacement data and the vehicle direction information to calculate the next location. This technique is able to obtain the current position and the distance travelled since the last known position [7, 104]. It is generally used to overcome the non-existence of GPS signal, and can be used only for short period in VANET, since it can easily accumulate errors [33].

MC - it is a Markov model in which each state corresponds to an observable (physical) event [207, 175]. Markov models can use the movement history in cellular/mobile or vehicular networks, in which user/node movements are *mined* from geographical positioning system (e.g., GPS, Galileo, cellular) traces.

HMM - based on Markov model, like the MC, is a doubly embedded stochastic process with an underlying stochastic process that is not observable (it is hidden), but can only be observed through another set of stochastic processes that produce the sequence of observations [207]. MC and HMM are probabilistic models that are built assuming that given a sequence of states, the probability of a next

state depends only on the current state and not on any other previous states (a Markov property). This approach defines a memoryless property for creating the model. The Markov property, however, can be extended by considering more previous state (e.g., the work by [89]), which is dubbed the *order* of the Model.

ANN, are Machine Learning (ML) based approaches [32, 299]. Neural Networks are non-linear processes and have capacity of learning and generalization. Moreover, they present efficient hardware implementation [113]. Therefore, they are more suitable for mobility prediction in VANET.

Data mining - is the science of extracting useful knowledge from huge data repositories [56]. For mobility prediction, it is based on different knowledge, such as road topology information, user behavior, and movement parameters [298].

Filtering - which groups schemes such as: a) the Particle Filters (PF) [218], which are sequential Monte Carlo methods based on the weight representation of probability densities of any given state model [18]; b) KF [124, 267], an efficient recursive filter that estimates the state of a linear dynamic system from a series of noisy measurements; and c) KF variants (i.e., EKF [121], and Unscented Kalman Filter (UKF)), proposed to deal with non-linear dynamics and non-linear measurement models.

Table 3: Summary of different mobility prediction approaches (based on [298, 32])

Algo-rithm	Training	Main feature	Accuracy [range %]	Prediction output
DR	Not required	Good for short time window; Accurate for linear mobility pattern, but highly related to the correction frequency	Not available	Future location
KF	Not required	Good for short time window; Accurate for linear mobility pattern, but highly related to the correction frequency (For non-linear systems, EKF or UKF can be used)	Not available	Future location
PF	Required	Good for short time window; Accurate for non-linear mobility pattern	Not available	Future location
ANN	Required, huge training complexity	Good for long time window; Accurate for non-linear mobility pattern	[74-92]	Future location and Trajectory
MC	Required; Prediction based on short memory principle (n-order high, the model can "overfit" the training sequence, if low, relationship between symbols cannot be captured)	Good for long time window; Low performance in complicated environment; Accurate for non-linear mobility pattern	[25-72]	Future location and Trajectory
HMM	Required; Based on Markovian approaches, can achieve better accuracy than MC; Requires more dedicated training than MC	Good for long time window; Accurate for non-linear mobility pattern	[68-82]	Future location and Trajectory

The study by [18] presents a performance evaluation and subsequent comparison of the following algorithms: KF, EKF, UKF, Alpha-Beta-Gamma (ABG) filter [163] and PF. The study in [86] presents the comparison of ANN- and a KF-based algorithm. The study by [298] presents a comparative description of different algorithms (e.g., MC, HMM, ANN, Bayesian network, KF, and data mining based) in mobility prediction. The comparison considers: a) the main features and accuracy; b) prediction output; c) prediction metrics; d) required information by the algorithm; as well as e) the main distinction of these algorithms.

The main conclusion from these studies is that ANN-based solutions present better performance for both linear and non-linear systems, but at a cost of much more training and much powerful computational processing. Moreover, although ANN-based solutions can obtain good prediction results, they are generally difficult to converge and are prone to local minimums [299]. PF-based solutions, designed for non-linear systems, present better performance than KF-based solution, but at a cost of more dedicated training (i.e, high computational cost). The UKF-based solutions perform better than the solutions based on the original KF or EKF solution. The EKF solutions are designed for locally linear systems. See Table 4 for a brief comparison of the characteristics of filter based solutions. For systems with higher non-linear properties, the alternatives to EKF can be UKF and PF, which, instead of approximating a non-linear function, approximate the probability distribution (sigma points for UKF, or particles for PF—for any distribution, not only Gaussian [55]).

Table 4: Comparison of filter-based estimator approaches ([55])

Algorithm (Estimator)	Model	Assumed Distribution	Computational Cost
KF	Linear	Gaussian	Low
EKF	Locally linear	Gaussian	Low (For analytically computed Jacobians) Medium (For numerically computed Jacobians)
UKF	Non-linear	Gaussian	Medium
PF	Non-linear	Non-Gaussian	High

MC- and HMM-based solution are designed to work with non-linear systems. They require training but are less computationally complex than the ANN-based solutions. Besides, HMM solutions can enable the mobile node to learn the environment and update the information itself, increasing their performance further. Therefore, the HMM-based solutions present better performance than the MC-based ones, but at a cost of more dedicated training.

The mobility prediction algorithms proposed in this thesis are based on KF/EKF and HMM. Therefore, we present in the next sections the details of these approaches.

2.4.1 Kalman Filter (KF)

The KF is a set of mathematical formulations implementing an efficient prediction-correction estimator, which minimizes the estimated error covariance under certain assumptions (i.e., system/functions linearity and Gaussian distributions). The KF estimation process is based on a feedback control mechanism. It first estimates the process state at a given future discrete time and then incorporates feedback in the form of noisy measurements (noisy). Thus, the mathematical equations for filtering are divided into two groups: 1) time update (predictor equations) — which project the current state and error covariance forward to obtain the *a priori* estimates for the next time step; and 2) measurement update (corrector equations) — which incorporate new measurements to correct the *a priori* estimate and obtain an improved *a posteriori* estimate [267].

The KF state estimator is defined by the linear state transition and measurement functions, as presented in Equations 2.1 and 2.2, respectively.

$$x_k = Ax_{k-1} + Bu_k + w_{k-1} \quad (2.1)$$

$$z_k = C_k x_k + v_k \quad (2.2)$$

The KF estimation process iteratively performs a prediction step followed by a correction step to determine the filter states and reduce prediction uncertainty. The steps of this iterative process are shown as follows. Equations 2.3 and 2.4 correspond to the time update (prediction) process, while Equations 2.5, 2.6, and 2.7 correspond to the measurement update (correction) process.

KF Time Update Equations (Prediction Step)

$$\hat{x}_k^- = A\hat{x}_{k-1} + Bu_k \quad (2.3)$$

$$P_k^- = AP_{k-1}A^T + Q \quad (2.4)$$

KF Measurement Update Equations (Correction Step)

$$K_k = P_k^- C^T (C P_k^- C^T + R)^{-1} \quad (2.5)$$

$$\hat{x}_k = \hat{x}_k^- + K_k (z_k - C\hat{x}_k^-) \quad (2.6)$$

$$P_k = (I - K_k C) P_k^- \quad (2.7)$$

where:

$\hat{x}_k^-$ – *a priori* state estimate at time k (before incorporating measurements)

$\hat{x}_k$ – *a posteriori* state estimate at time k (after incorporating measurements)

A – state transition matrix

B – control input matrix

u_k – control input at time k

C_k – measurement matrix at time k

K_k – Kalman gain at time k

z_k – measurement at time k

R – measurement noise covariance

Q – process noise covariance

P_k – error covariance matrix at time k

w_k – process noise (zero-mean, covariance Q)

v_k – measurement noise (zero-mean, covariance R)

2.4.2 Extended Kalman Filter (EKF)

The KF described in Section 2.4.1 is defined on the assumption that the system on which the estimation is being performed is linear. If the system does not meet the aforementioned assumption, the performance of KF is not guaranteed, and the KF may not converge. In this case, the EKF, which linearizes the model function, may be used. After linearizing the model function, the EKF state estimator can be defined by the nonlinear state transition and the measurement function (Equation 2.8 and 2.9)

$$x_t = f(x_{t-1} + u_t) + w_t \quad (2.8)$$

$$z_t = g(x_t) + v_t \quad (2.9)$$

After the linearization, the prediction process and the correction process, described by the Equations 2.3, 2.4, 2.5, 2.6 and 2.7, are now given by the following Equations 2.10, 2.11 (for time update process), and Equations 2.12, 2.13 and 2.14 (for measurement update process), as presented in [268].

EKF time update equations (prediction process)

$$\hat{x}_t^- = f(\hat{x}_{t-1}, u_t, 0) \quad (2.10)$$

$$P_t^- = A_t P_{t-1} A_t^T + W_t Q_{t-1} W_t^T \quad (2.11)$$

EKF measurements update equations (correction process)

$$K_t = P_t^- C_t^T (C_t P_t^- C_t^T + V_t R_t V_t^T)^{-1} \quad (2.12)$$

$$\hat{x}_t = \hat{x}_t^- + K_t (z_t - g(\hat{x}_t^-, 0)) \quad (2.13)$$

$$P_t = (1 - K_t C_t) P_t^- \quad (2.14)$$

Where A_t , W_t , H_t , V_t , are Jacobian matrices of partial derivative of:

$$A_t = \frac{\partial f}{\partial x}(\hat{x}_{t-1}, u_t, 0) - f \text{ in respect to } x \text{ at time } t$$

$$W_t = \frac{\partial f}{\partial w}(\hat{x}_{t-1}, u_t, 0) - f \text{ in respect to } w \text{ at time } t$$

$$H_t = \frac{\partial h}{\partial x}(\hat{x}_t, v_t, 0) - h \text{ in respect to } x \text{ at time } t$$

$$V_t = \frac{\partial h}{\partial v}(\hat{x}_t, v_t, 0) - h \text{ in respect to } v \text{ at time } t$$

For systems with high nonlinear proprieties, the alternatives to EKF can be the UKF, and the PF, which instead of approximating a nonlinear function, approximate the probability distribution (Sigma points for UKF, or particles for PF - for any distribution, not only Gaussian [55]).

2.4.3 Hidden Markov Model (HMM)

A HMM is a doubly embedded stochastic process with an underlying non-observable stochastic process (i.e., it is hidden), which can only be observed through another set of stochastic processes that produce the sequence of observations [207]. Next, the HMM is presented, as explained in [207].

Two kinds of states are presented in HMM: the hidden and observable states. Moreover, two main parameter sets of the Markov Model (MM) (i.e., the transition probabilities - A , and the initial probabilities - π) are used along with a third new parameter set (i.e., the observation/emission symbol probability - B), which provides the relationship between the state (hidden state) and the observation, as illustrated by Figure 7.

The following is the nomenclature used to describe the HMM. We include here the specific characterization adopted for this thesis.

T - length of the observation sequence. In our model it is equivalent to the number of the historic mobility traces for each vehicle.

N - number of states. It is equivalent to the number of road intersections and the considered Staying Place (SP) for each vehicle.

M - number of observation symbols. It is equivalent to the number of Node Mobility Status Information (NMSI) points. That is, each time a NMSI is registered, a location is registered and this location plus the timestamp is used as an individual observation symbol. Several NMSI will be registered along the road segments.

$V = \{v_1, v_2, \dots, v_M\}$ - set of discrete observation symbols. It is equivalent to each discrete NMSI point. Specifically, each observation symbol (v_i , with $i \leq M$) is formed by a geographical location (i.e., by latitude and longitude values), current speed (i.e., in 2D values x, y), the node ID, and a timestamp.

$Q = \{q_1, q_2, \dots, q_N\}$ - states in the model. Each q_i , with $i \leq N$, corresponds to a specific intersection or a registered SP location (e.g., each of the x_i , or the destinations/initial location).

The characterization of an HMM is given by the following elements:

1. $Z = \{Z_1, Z_2, \dots, Z_T\}$, which represents the hidden state sequence of the model, from the set of symbols Q , with $Z_i \in Q$;
2. $O = \{O_1, O_2, \dots, O_T\}$ represent the observation sequence, from the set of symbols V , $O_i \in V$;
3. The state transition probability distribution, which represent the probability of moving from a hidden state Z_i to Z_j , and is given by an $N \times N$ matrix, $A = \{a_{ij}\}$;

$$a_{ij} = P[Z_j \text{ at } t + 1 | Z_i \text{ at } t], \quad \sum_{j=1}^N a_{ij} = 1 \quad \forall i \quad (2.15)$$

Which means that, for a 1st order Markov chain, the probability of a particular hidden state only depends on the previous state - *the Markov assumption*.

4. The observation symbol probability distribution in state Z_j , as an $N \times M$ matrix, $B = \{b_j(k)\}$, where

$$b_j(k) = P[v_k \text{ at } t | Z_j \text{ at } t], \quad \sum_{j=1}^N b_j(k) = 1 \quad \forall k \quad (2.16)$$

Which means that the probability of an observed output v_i , depends only on the hidden state that produced the observation sequence - the *output independence*.

5. The initial state distribution (initial probability array) $\pi = \{\pi_i\}$, where

$$\pi_i = P[Z_i \text{ at } t = 1], \quad \sum_{i=1}^N \pi_i = 1 \quad (2.17)$$

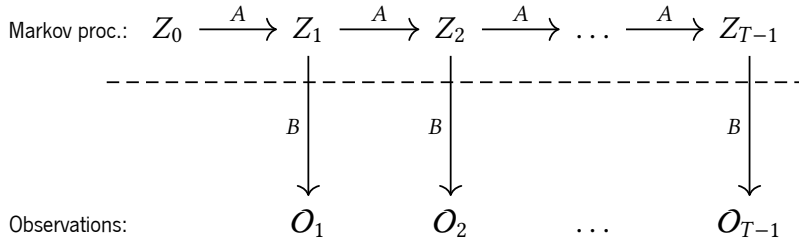


Figure 7: The Hidden Markov Model

A complete characterization of an HMM requires the definition of two model parameters (i.e., N and M), specification of observation symbols, and the three probability measures, A , B and π . The complete definition of HMM is given by Equation 2.18

$$\lambda = (N, M, A, B, \pi) \quad (2.18)$$

Or, by its compact definition, Equation 2.19.

$$\lambda = (A, B, \pi) \quad (2.19)$$

HMM is characterized by three fundamental problems:

Problem 1 (Likelihood): Given an HMM $\lambda = (A, B, \pi)$ and an observation sequence $O = O_1 O_2 \dots O_T$, determine the likelihood $P(O|\lambda)$, i.e., how well the model matches a given observation sequence.

Problem 2 (Decoding): Given an HMM $\lambda = (A, B, \pi)$ and an observation sequence $O = O_1 O_2 \dots O_T$, discover the best hidden state sequence Z . Although other methods might be used, *Viterbi* algorithm [207], a kind of dynamic programming, which aims at finding the most likely state sequence $Z = Z_1 Z_2 \dots Z_T$ associated with a given observation sequence $O = O_1 O_2 \dots O_T$, is the mostly used algorithm. Equation 2.20 is iteratively computed to calculate and store the maximum probability to move from state q' to q , and thus recursively predict the most likely state sequence.

$$\delta_q(t) = \max_{1 \leq q' \leq N} [\delta_{q'}(t-1) * a_{q'q}] * b_q(O_t) \quad (2.20)$$

Problem 3 (Learning): Given an observation sequence $O = O_1 O_2 \dots O_T$, learn and adjust the model parameters $\lambda = (A, B, \pi)$, in order to maximize $P(O|\lambda)$. The solution to this problem can be seen as the training of the model to better fit the observation sequence. The *Forward-backward* algorithm [37], is the most commonly used algorithm to estimate the parameters [146].

The *Baum-Welch* [38], a *Forward-backward*-based algorithm, is adopted to estimate and train the model parameters (A , and B) with the existing mobility traces.

The training process, is described by the *Baum-Welch* algorithm and the following set of variables [207]:

- a) The forward probability $\alpha_i(t) = P(O_1, O_2, \dots, O_t, Z_i = t|\lambda)$ - which can be computed by the *Forward-backward* algorithm, is the probability of the partial observation sequence (up to time t), and the state Z_i at time t ;
- b) The backward probability $\beta_i(t) = P(O_{t+1}, O_{t+2}, \dots, O_T | Z_i = t, \lambda)$ - Which can be computed by *Backward probability* algorithm, is the probability of seeing the observations from time $t + 1$ to the end, given that we are in state Z_i at time t ;
- c) $\xi_{i,j}(t) = P(Z_i = t, Z_j = t + 1 | O_1, O_2, \dots, O_T, \lambda)$ - the probability of moving from state Z_i at the time t to the state Z_j at the time $t + 1$, given the model λ and the observation sequence; and
- d) $\gamma_i(t) = P(Z_i = t | O_1, O_2, \dots, O_T, \lambda)$ - the probability of being in state Z_i at the time t , given the observation sequence and the model λ .

In summary, the HMM training process follows the steps presented next and is iterated until it converges or a previously defined threshold is met.

1. Initialize $\lambda = (A, B, \pi)$
2. For each training sequence (i.e., the mobility trace history), compute $\alpha_i(t)$, $\beta_i(t)$, $\gamma_i(t)$, $\xi_{i,j}(t)$ - using the *Forward Probability* algorithm and the *Backward Probability* algorithm to iterate and calculate $\alpha_i(t)$, and $\beta_i(t)$, respectively.
3. Re-estimate $\lambda = (A, B, \pi)$ - using the *Baum-Welch* algorithm.
4. If $P(O|\lambda)$ increases, goto 2.

2.5 Summary

This chapter presented the theoretical background of the technologies used by the present thesis. An overview of the main technologies: VANET and its proprieties and characteristics, NDN architecture, and the main mobility prediction theories addressed in the work. VANET has been foreseen as an important technology for the development of ITS. The proprieties and characteristics of NDN architecture, clearly put it as a potential architecture to overcome the challenges still faced by the IP-based network architecture. Mobility prediction algorithms have been seen as important addons in VANET, either to help on correcting the possible errors from the geographical system applications or to provide predictions where no geographical system applications are available. The synergies from these technologies can bring invaluable opportunities to ITS.

NDN-based VANET - State-of-the-art

This chapter aims at presenting the state-of-the-art on the matter addressed in this thesis. The state-of-the-art is assessed by means of a SLR performed and published in [232].

The chapter is organized as follows: Section 3.2 presents the related work. Section 3.3 presents the methodology that guides the development of the SLR. Next, Section 3.4 presents the research questions, Section 3.5, the selected sources for searching data, Section 3.6 presents the inclusion and exclusion criteria. Section 3.7 presents the included primary studies and Section 3.8 presents the data extraction, and synthesis of the selected papers. Section 3.9 presents the data analysis and the interpretation of the results. Section 3.10 presents the main improvements applied to VANET by using NDN. Later, before presenting the conclusions in Section 3.12, we decided to include an additional section on forwarding - Section 3.11. The data assessment for the aforementioned section was performed lately and followed a different methodology. This was done this way having in consideration that there were several surveys already performed for forwarding, when the SLR was performed, as already stated.

3.1 Context

As previously referred, current communications are content-centric, i.e., the main element for communication is the content, instead of the host. This fact is a major shift in communications by computers and poses new challenges to IP-based networks. This shift in technology demanded the development of new architectural paradigms capable of supporting natively the new reality. Among the several proposed Content-Centric architectures (see Section 2.2), NDN, a particular implementation of CCN, has been considered as the most promising solution [252, 301]. As previously highlighted, there is an enormous potential of advantages on realization of NDN-based VANET. The combination of both technologies is an exciting topic and one on which recent developing efforts are being put forward.

A quick web search has shown that although several solutions were proposed for improving VANET by NDN, still a scarce number of studies proposed solutions with focus on routing. Besides routing, forwarding, caching, security and privacy are other main topics researchers in ICN are putting their effort on. The need to access the state-of-the-art and, therefore, extract the main gaps from the proposed

solutions on improving NDN-based VANET on the aforementioned topics, motivated the development of this survey.

3.1.1 Objectives

The purpose of this study is to survey the existing contributions on the feasibility of improving the performance of VANET using NDN. There exist several similar studies but as we explain later, they focus in forwarding or when including other topics such as routing, they treat forwarding and routing interchangeably. Differently, this study limits its focus on the solutions for improving VANET by NDN-based routing (not forwarding), caching and security. From the findings, it also proposes the possible lines for further development in the aforementioned topics. This purpose is attained by performing a research based on a SLR, which covered a period from 2010 to 2021.

We highlight that considering the specific differences of VANET (i.e., the higher speed, with predictable movements; the highly dynamic network topology with frequent partitions; and the less restrictions related to energy consumption, processing power, and memory availability) from MANET, our study only focus on VANET.

3.1.2 Main contributions

Although several surveys have been performed on the improvement of VANET using NDN, the topics likely to be covered were not completely or even individually fully surveyed. Thus, this work proposed to close the existing gaps. Routing and forwarding, in NDN, are two completely separated planes and likely to be developed separately [302]. Now, considering that the majority of related surveys treat these two topics indiscriminately, a survey focusing on bringing the real state-of-the-art of routing, in order to provide a clear figure of what has been done, is still missing. More precisely, we intent to survey solutions applying routing protocols for NDN-based VANET not those using pure forwarding (i.e., only the broadcast mechanisms). Routing solutions explicitly manage FIB.

In summary, our main contributions include the following: 1) We highlight the need to differentiate the control and data planes in NDN, specifically in case of VANET; 2) Given that several surveys focusing on topics including forwarding has been performed, we survey the specific solutions for NDN-based routing. Note that although we scrutinized each of the related work, extracted and discriminated forwarding from routing solutions, none of the related surveys present any routing proposal developed after the year 2019; 3) Related work surveyed the topic on caching, however we felt the surveys were not sufficiently comprehensive (several solution in the surveyed period were left apart), thus we closed this gap complementing this topic; 4) Surveyed solutions for security are scarce, thus, we also close this gap; and 5) We try to present more precisely the open challenges and future roadmap for tackling them.

Table 5 (Section 3.2) shows the summary of the related surveys and their limitations/gaps, which we try to close within this study.

3.2 Related surveys

Several studies (e.g., [277, 20, 77, 97, 60, 10, 29, 72, 238]) aiming at holistically leverage the characteristics of CCN/NDN for VANET have been performed. For specific improvements (i.e., specific for routing, forwarding, caching, privacy, security, and so forth) related references can be found in the many existing reviews and surveys, as presented in Table 5. These studies can be complemented by several surveys and reviews performed on general CCN/NDN (not specific for VANET) issues, such as the naming and routing [28, 35], caching and forwarding [116, 297], transport [61], mobility [305, 140], and security [57, 229].

The following studies are related surveys. The coverage of these papers ranges from 2014 to 2021, and survey solutions proposed from 2010 to 2020.

Authors in [25] presented guidelines and highlighted the points of strength and weaknesses on the applicability of CCN principles to wireless networks. The aforementioned study reviewed the state-of-the-art on naming, routing and forwarding, transport, caching, and security applied to MANET, VANET and Wireless Sensor Networks (WSN). This survey is more than 5 years old.

In [46], concepts such as naming, routing, security, and caching including the research challenges applied to Vehicular Content Centric Networks (VCCN) are presented. This survey is also more than 5 years old, and it is limited in number of surveyed studies.

The surveyed papers by [286] are also scarce, focusing only on forwarding and missing out several relevant studies within the considered period. This survey is also more than 5 years old.

In [21], ICN for connected vehicles are surveyed. This study also presents the future perspectives on topics including naming, routing, forwarding, security and caching. Although covering a large spectre of topics, this study does not go deeper on surveying any of these topics. Thus, leaving out several relevant solutions within the surveyed period. This survey is more than 5 years old.

Aiming at providing references and guidelines for readers with interest on the ICN based MANET, authors in [159] presented an comprehensive overview on this topic and surveyed the proposed solutions including for VANET and Delay Tolerant Networks (DTN). This study also misses out several studies on the surveyed topics, mainly on routing and caching.

Authors in [140] presented a comprehensive survey on Vehicular Named Data Networking (V-NDN) characteristics, where topics such as mobility, routing, caching, forwarding, and security are presented. Although referred, the topic on routing in contrast with forwarding is not sufficiently surveyed.

Authors in [135] presented the current status and future challenges related to the internet of vehicles. The study is based on a survey including the following topics: forwarding, mobility, security, and caching. This study misses out the topic on routing and several relevant studies on the selected topics.

Authors in [240] surveyed the forwarding strategies and like [25], the review is based on MANET, VANET, and WSN. Additionally, the study presents the future challenges regarding the forwarding in wireless environment. Similar effort was done by [8], which also focused the survey on forwarding strategies

but only for vehicular networks. Both studies cover relatively the same period and as referred only surveyed the forwarding strategies.

Authors in [58] performed a comprehensive survey focusing only on caching. This study was published relatively in the same period when the initial data extraction was being performed for the present study. Anyway, the results presented in that paper are complemented by the results produced here. In [209], a brief survey in content naming schemes is performed.

Table 5 presents the summary of the related works.

Table 5: Summary and comparison of a representative related survey papers

Ref.	Year	Period	Cited	Surveyed papers by topic						Total	Main limitations (related to this survey)
				Naming	Forward.	Routing	Caching	Mobility	Security		
[25]	2014	[2010-2014]	065	02	05	01	01	00	01	09	Missing relevant literature;
[46]	2015	[2010-2014]	020	00	04	01	00	00	00	05	Limited on routing. Caching and security not included;
[303]	2015	[2006-2014]	055	00	00	00	16	00	00	16	Only surveys caching and misses relevant literature;
[21]	2016	[2010-2014]	015	06	04	01	05	00	02	18	Missing relevant literature;
[286]	2016	[2013-2015]	024	00	03	00	00	00	00	03	Only surveys forwarding and misses relevant literature;
[311]	2016	[2010-2016]	048	03	06	02	04	00	00	19	Missing relevant literature;
[305]	2016	[2012-2015]	028	00	00	00	00	13	00	13	Only surveys mobility and misses relevant literature;
[159]	2017	[2010-2016]	136	04	18	03	02	00	01	28	Missing relevant literature;
[225]	2017	[2015-2017]	030	01	11	00	00	00	00	12	Missing relevant literature;
[135]	2019	[2010-2019]	055	00	06	00	09	00	03	18	Routing not included; Missing relevant literature on caching;
[140]	2020	[2010-2018]	199	11	40	05	07	07	05	75	Relevant literature are missing for routing and caching;
[240]	2020	[2010-2019]	082	00	23	00	00	00	00	23	Caching, routing and security not included;
[8]	2020	[2010-2020]	065	00	34	00	00	00	00	34	Caching, routing and security not included;
[58]	2020	[2010-2020]	174	00	00	00	31	00	00	31	Only caching but missing some relevant literature;
[209]	2021	[2014-2019]	037	07	00	00	00	00	00	07	Only naming schemes but missing some relevant literature;
Our	2022	[2010-2021]	197	00	00	34	48	00	12	94	Comprehensive but only specific for the selected topics.

As shown in the table, the forwarding is the most covered topic by the existing surveys. Although some studies referred routing, in fact they only cover the forwarding or when including routing they do it indiscriminately from forwarding. Thus, the need to separately survey routing. Apart from the study in [58], which was published relatively in the same period when the extraction of data for the present work was being performed, all other related papers present a relatively light surveying on caching.

In the TCP/IP architecture, data plane performs forwarding based on the decisions given by the control plane. In CCN based architectures, such as NDN, the forwarding plane is stateful and adaptive and capable of performing in local networks without the routing plane [25, 301]. Although both planes are explicitly separated and likely to be developed independently, the aforementioned fact apparently makes that the relative difference in terms of importance of both planes seems irrelevant, making the majority of literature treat these two topics indiscriminately. Differently from the related surveys, we discriminate the forwarding from routing as, in fact, these are two different topics. Given that most of the related work have already dealt with forwarding, we focus our effort on routing.

More precisely, besides the other topics (i.e., caching and security, which are scarcely surveyed by most of the related surveys) this work intent to survey solutions applying routing protocols for NDN-based

VANET, apart the studies that only propose forwarding strategies, i.e., are broadcast-based. Given that literature [96, 97] consider infeasible to run a routing protocol in VANET, it is interesting to survey what proposals and what results are given by those studies proposing routing, compared to the broadcast-based forwarding solutions.

3.3 Methodology

This section presents the SLR methodology used to conduct the present study. Authors in [43] and [245] define SLR as a research methodology used to collect, map and evaluate the existing scientific knowledge on a specific topic. Based on SLR findings, the researcher can specify a research question to further develop the body of knowledge. This type of review follows a well defined and strict sequence of steps, according to a protocol developed *a priori*.

The present SLR is based on the guidelines proposed by [245] and [43], and refined by [147]. The steps to follow for performing the SLR are summarized in Figure 8 and explained as follows:

1. Planning Phase - can be thought as the protocol definition phase where the methods used to undertake the review are specified. This activity is fundamental to reduce the possibility of researcher bias. The following activities are developed: a) Research question formulation - the research questions are raised and the objectives of the review are defined; and b) Selection of sources - selection of the sources where primary studies will be searched;
2. Execution Phase - this phase can be divided into the following activities: a) selection of primary studies from the sources; b) data extraction; c) data synthesis; and
3. Data analysis and interpretation phase - data are analysed and interpreted in this phase.

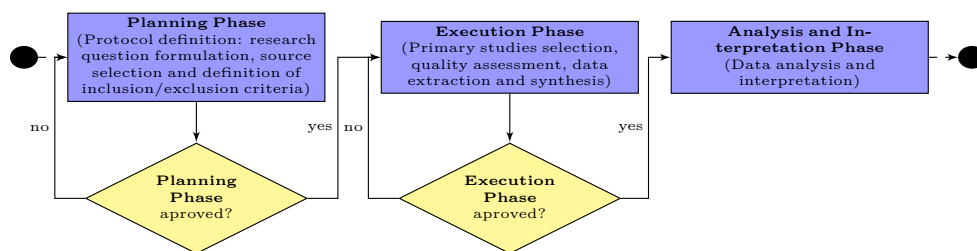


Figure 8: Systematic literature review process (adapted from [43])

3.4 Research Questions and Search Strings

To attain the aforementioned objectives, the following research questions (RQ1, RQ2, and RQ3) are formulated, based on general open challenges for ICN [114, 197, 151, 222]:

RQ1 - How can Named Data Networking routing be applied to improve VANET?

RQ2 - How can Named Data Network caching be applied to improve VANET?

RQ3 - How can Named Data Network based VANET security, privacy and trust issues be tackled?

From the research questions, *keywords* and *synonyms* were extracted (see Table 6). These keywords are concatenated with boolean operators - *AND* and/or *OR* - to form the *search strings* (see Table 7 and Table 8). The search strings are used to gather the primary studies from web search engines. Initially, the search and selection are based on all document metadata, then are refined by reading the title, keywords, abstract and taking an overview of the full paper.

Table 6: Keywords and Synonyms

Keywords	Synonyms
Named Data Networking	Named Data Network; NDN; Information-Centric Network; ICN; Content-Centric Network; CCN
Vehicular Ad hoc Network Caching	VANET, Vehicle to Everything, Vehicular Social, Internet of Vehicle, Internet of Vehicles (IoV), V2X, V2V, V2I Cache; Content Store

Table 7: Composed synonyms strings per keyword

Composed String (CStr) per keyword	
CStr1	(("Named Data" AND network*) OR "Information-Centric" OR "Content-Centric" OR NDN OR ICN OR CCN)
CStr2	(Cache OR caching OR "Content Store")
CStr3	("Vehicular Ad hoc" OR "Vehicle to Everything" OR "Vehicular Social" OR "Internet of Vehicle" OR IoV OR V2X OR VANET*)

Table 8: Strings used to search for primary studies

Search Strings	
RQ1	CStr1 AND Routing AND CStr3
RQ2	CStr1 AND CStr2 AND CStr3
RQ3	CStr1 AND (Security OR Privacy OR Trust) AND CStr3

3.5 Selected sources and studies

To conduct this study the following online bibliographical databases were selected: Scopus [221], Web of Science (WoS) [264], ACM Digital Library (ACM DL) [4], IEEE Xplore [115], and Google Scholar (GS) [93] - an online search engine (see Table 9). The query interface of these databases and search engine is slightly different from each other, thus a slight adaptation of the search strings was performed for each database or search engine. For instance, a search string composed by three keywords without any explicit boolean operator is interpreted - by GS, Scopus, and IEEE Xplore - as being AND'd. The same sequence is interpreted by ACM DL as being OR'd. The search strings presented in Table 8 are already adapted to be used on any of the selected source of information without further modification.

Scopus, GS, and WoS are the primary sources used to perform the search. We expect to find, using these search engine and databases, all the studies related to the topics. Nonetheless, the other sources: ACM DL and IEEE Xplore, were also searched and their results compared to the obtained with the former search engine and databases.

Table 9: Literature sources

Online databases	Scopus, WoS, ACM DL, IEEE Xplore
Online Search engine	GS

3.6 Inclusion and exclusion criteria

To select the correct set of studies related to the researched problem, the inclusion and exclusion criteria must have previously been defined and can be refined along the execution phase. Table 10 presents the inclusion and exclusion criteria defined for this work.

Table 10: Inclusion and exclusion criteria

Type	Criteria
Inclusion	Research papers published either in Conferences or Journals. - If a study has conference and journal article versions, the latter is selected; - If a study has many published versions, only the newest and the most complete version is selected; - Only papers published in Q1 and Q2 conferences and journals are included;
Inclusion	Research papers published between 2010 and 2021.
Exclusion	Studies not specifically addressing elected topics in NDN-based VANET.
Exclusion	Studies only performing surveys and reviews.
Exclusion	Studies not written in English.
Exclusion	Studies addressing the selected topics but with focus on MANET instead of VANET.

3.7 Inclusion of primary studies

The search was last updated in January 2022 considered all the metadata for the searched papers. The results are shown in Table 11. A total of 9077, 5300, and 4680 results were found for RQ1, RQ2, and RQ3, respectively.

After this first selection, herein considered as the 1st stage selection which gave a total of 19057 articles, a further selection was performed in two more stages.

In the 2nd stage selection, the papers are evaluated by their title and keywords. After the selection, duplicated papers within each query were eliminated. This stage produced 987 papers still to be scanned.

In the 3rd stage selection, the papers were scrutinized by their abstract and by an overview reading. Some selected papers from a given query were found to belong to other queries. The studies were, then, reorganized within the correct query and the duplicated studies among the queries were eliminated. This stage resulted in a more refined selection of 94 studies.

Table 11 shows the number of selected studies for each selection stage. The last column shows the percentage of the selected studies per topic, from the total of 94 studies. The papers selected after this process were full read and included in the data extraction process, which is the next step within this execution phase.

Table 11: Number of selected studies

	WoS	Scopus	GS	ACM	IEEE	Selection Stages			%
						1 st	2 nd	3 rd	
RQ1	083	2211	6570	107	106	9077	431	34	36.17
RQ2	103	0645	3860	610	082	5300	350	48	51.06
RQ3	058	2437	2030	088	067	4680	206	12	12.77
Grand total						19057	987	94	100.0

Table 12 presents the distribution of the primary studies, per sources. As this table shows, only 6 of the selected studies were found on the ACM DL and all studies were found using the GS online search engine.

Table 12: Distribution of the selected primary studies

Study	Year	Research question			Source of primary studies				
		RQ1	RQ2	RQ3	WoS	Scopus	GS	ACM	IEEE
[29]	2011	✓	-	-	✓	✓	✓	✓	-
[292]	2013	✓	-	-	✓	✓	✓	-	✓
[293]	2013	✓	-	-	✓	✓	✓	-	✓
[97]	2014	-	✓	-	✓	✓	✓	✓	-
[205]	2014	-	✓	-	✓	✓	✓	-	✓
[277]	2014	✓	-	-	-	✓	✓	-	-
[42]	2015	-	✓	-	✓	✓	✓	-	-
[157]	2015	-	✓	-	✓	✓	✓	-	✓
[73]	2016	-	✓	-	✓	✓	✓	-	✓
[100]	2016	-	✓	-	✓	-	✓	-	✓
[136]	2016	✓	-	-	✓	-	✓	-	✓
[174]	2016	-	✓	-	-	✓	✓	-	-
[263]	2016	✓	-	-	✓	✓	✓	-	✓
[26]	2016	✓	-	-	-	✓	✓	-	-
[204]	2016	-	✓	-	✓	✓	✓	-	-
[265]	2016	-	✓	-	-	✓	✓	-	-
[235]	2016	-	✓	-	-	✓	✓	-	-
[272]	2016	-	✓	-	✓	✓	✓	-	-
[1]	2017	-	✓	-	-	✓	✓	✓	-
[130]	2017	✓	-	-	✓	✓	✓	-	✓
[128]	2017	✓	-	-	✓	✓	✓	-	✓
[129]	2017	✓	-	-	✓	✓	✓	-	✓

Continued on next page

Table 12 – continued from previous page

Study	Year	Research question			Source of primary studies				
		RQ1	RQ2	RQ3	WoS	Scopus	GS	ACM	IEEE
[180]	2017	-	✓	-	✓	✓	✓	-	✓
[307]	2017	-	✓	-	✓	✓	✓	-	✓
[102]	2017	✓	-	-	-	✓	✓	-	✓
[170]	2017	✓	-	-	-	-	✓	-	✓
[108]	2017	-	✓	-	-	-	✓	-	✓
[64]	2017	-	-	✓	✓	-	✓	✓	✓
[63]	2017	-	-	✓	✓	-	✓	✓	-
[304]	2017	-	-	✓	-	✓	✓	-	✓
[82]	2018	-	✓	-	✓	✓	✓	-	-
[84]	2018	-	✓	-	-	-	✓	-	-
[125]	2018	✓	-	-	✓	✓	✓	-	✓
[142]	2018	-	✓	-	-	✓	✓	-	✓
[164]	2018	-	✓	-	✓	✓	✓	✓	-
[179]	2018	-	✓	-	✓	✓	✓	-	✓
[190]	2018	-	✓	-	✓	✓	✓	-	✓
[280]	2018	-	✓	-	-	✓	✓	-	-
[67]	2018	✓	-	-	-	-	✓	-	✓
[250]	2018	-	✓	-	✓	-	✓	-	✓
[109]	2018	-	✓	-	-	-	✓	-	✓
[182]	2018	✓	-	-	-	-	✓	-	✓
[80]	2018	✓	-	-	✓	✓	✓	-	✓
[76]	2018	✓	-	-	-	✓	✓	-	✓
[74]	2018	✓	-	-	✓	✓	✓	-	✓
[53]	2018	✓	-	-	-	✓	✓	-	✓
[214]	2018	✓	-	-	✓	✓	✓	-	-
[81]	2018	✓	-	-	✓	✓	✓	-	-
[98]	2018	-	✓	-	✓	✓	✓	-	-
[111]	2018	-	✓	-	✓	-	✓	-	✓
[106]	2018	-	✓	-	✓	-	✓	-	✓
[145]	2018	-	-	✓	-	✓	✓	-	✓
[158]	2019	-	-	✓	-	-	✓	-	✓
[217]	2019	-	-	✓	-	✓	✓	-	✓
[282]	2019	-	✓	-	-	✓	✓	-	-
[79]	2019	-	✓	-	-	✓	✓	-	-
[110]	2019	-	✓	-	✓	✓	✓	-	✓
[126]	2019	✓	-	-	✓	✓	✓	-	✓
[285]	2019	-	✓	-	✓	✓	✓	-	✓
[141]	2019	-	✓	-	-	-	✓	-	✓
[193]	2019	-	✓	-	-	-	✓	-	✓
[275]	2019	✓	-	-	-	-	✓	-	✓
[254]	2019	✓	-	-	-	-	✓	-	✓
[178]	2019	✓	-	-	✓	-	✓	-	✓
[266]	2019	-	✓	-	✓	✓	✓	-	-
[99]	2019	-	✓	-	-	✓	✓	-	✓
[127]	2020	✓	-	-	✓	-	✓	-	-

Continued on next page

Table 12 – continued from previous page

Study	Year	Research question			Source of primary studies				
		RQ1	RQ2	RQ3	WoS	Scopus	GS	ACM	IEEE
[274]	2020	✓	-	-	✓	✓	✓	-	✓
[255]	2020	-	✓	-	-	-	✓	-	✓
[278]	2020	✓	-	-	-	-	✓	-	✓
[299]	2020	✓	-	-	-	✓	✓	-	-
[107]	2020	-	✓	-	-	-	✓	-	✓
[306]	2020	-	✓	-	-	-	✓	-	✓
[248]	2020	-	✓	-	✓	-	✓	-	✓
[281]	2020	-	✓	-	-	-	✓	-	✓
[24]	2020	-	✓	-	✓	-	✓	-	-
[153]	2020	-	-	✓	✓	✓	✓	-	-
[168]	2020	-	-	✓	-	-	✓	-	✓
[309]	2020	-	-	✓	-	-	✓	-	✓
[206]	2021	-	-	✓	✓	-	✓	-	✓
[27]	2021	✓	-	-	✓	-	✓	-	✓
[183]	2021	-	✓	-	-	✓	✓	-	-
[103]	2021	-	✓	-	-	✓	✓	-	-
[287]	2021	-	✓	-	-	✓	✓	-	-
[22]	2021	-	✓	-	-	✓	✓	-	-
[101]	2021	-	✓	-	✓	✓	✓	-	-
[230]	2021	✓	-	-	-	-	✓	-	-
[3]	2021	-	-	✓	-	✓	✓	-	-
[284]	2021	-	-	✓	-	-	✓	-	✓
[17]	2021	✓	-	-	-	-	✓	-	-
[228]	2021	✓	-	-	-	-	✓	-	✓
[123]	2021	-	✓	-	✓	✓	✓	-	-
[59]	2021	-	✓	-	✓	-	✓	-	✓

Figure 9 shows the distribution period of the selected studies. The selection of studies was performed between 2010-2021, which was chosen considering that the development of NDN architecture started around 2009 and the research aiming at integrating NDN with VANET was started even later. As the figure shows, the development of work aiming at integrating NDN with VANET begun around 2011 with its intensity between 2016 and 2021. It is also shown that among the three selected topics, the majority of studies is being developed for caching in NDN-based VANET (51.06% of all the 94 selected studies).

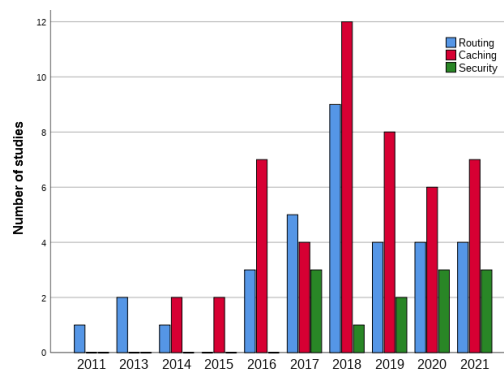


Figure 9: Year of publication of the selected studies, per area

3.8 Data extraction and synthesis

After the selection of the primary studies the data extraction process was performed. To facilitate this process a database sheet was produced. The collected parameters are explained in the following sections, in the following order: routing in Section 3.8.1, caching in Section 3.8.2, and security and privacy issues in Section 3.8.3. We only present extraction and synthesis in this section. Analysis and results interpretation are presented in Section 3.9.

3.8.1 Research Question (RQ1) - Routing

For routing scheme classification, we use the combination of taxonomy and classification proposed in [122, 78, 293]. Table 13 presents the extracted data, based on the aforementioned parameters for RQ1. The first three columns are auto explanatory. The mode of operation indicates the type of proposed protocol, as topology-based, geographic/position-based, based on opportunistic encounters, or cluster-based, as explained in Section 4.4.2. The type of application column indicates whether the VANET application is location dependent (e.g., active safety and efficiency related applications) or location independent (e.g., entertainment applications). When not explicit in the study we considered that the solution is designed for both the location-dependent and location-independent content.

The following are the synthesis of the proposed solutions for NDN-based routing.

In [277] a new vehicular information network architecture is proposed. The proposal includes a naming scheme designed to allow data aggregation and interest segregation, and a proactive position-based routing where the location information is the name of its geographical position. The location information and its correspondent next locations is used in each node to decide which path to forward the content.

Authors in [128] proposed a routing algorithm based on initial broadcast of Interest for content discovery. Subsequent requests are retrieved by unicast paths created by the Data packet on the downstream. The Media Access Control (MAC) address is chosen as the unambiguous node identifier and is added as a new field in the NDN packets. The proposal uses periodic broadcasts to discover new paths and new content sources. In [129], the same authors modify the previous proposal now extracting the MAC addresses from the NDN strategy layer. In order to improve the retrievability of information from vehicles and save network resources, the same authors propose in [125], a V2I communication architecture based on NDN, which is then extended to support V2V. For content discovery, the RSU broadcast beacon messages endlessly. The content sources respond the beacon messages by announcing their content prefixes. Also based on MAC addresses to identify the nodes, a geographical aware routing protocol - eGaRP, is proposed in [126] by the same authors. The solution assumes each vehicle has a set of directional antennas used to set unicast transmission to forward the messages in a specific direction after a route has been established. Current geographical coordinates (from a GPS device) of a given vehicle are stored in PIT and FIB, and are used by each node to calculate the angle between its own position to the node that should receive the packet to transmit. In [127] the details of eGaRP, including its performance

Table 13: Extracted data from primary studies for RQ1 - Routing

Study	Model		Mode of operation				Type of appl. (Location-)		
	V2V	V2I	Topology		Geographic	Opportunistic	Clustering	Dependent	Independent
			Proactive	Reactive					
[29]	✓	-	-	✓	-	-	-	✓	-
[292]	✓	-	-	✓	✓	✓	-	-	✓
[293]	✓	-	✓	-	✓	✓	-	✓	✓
[277]	✓	✓	✓	-	✓	-	-	✓	-
[136]	✓	✓	✓	-	-	-	-	✓	✓
[263]	✓	-	-	✓	-	-	-	✓	✓
[26]	✓	-	-	✓	-	-	-	✓	✓
[130]	✓	✓	-	✓	-	-	-	✓	✓
[128]	✓	-	✓	-	-	-	-	✓	✓
[129]	✓	✓	✓	-	-	-	-	✓	✓
[102]	✓	-	-	✓	-	-	-	✓	✓
[170]	✓	-	✓	-	-	-	-	✓	✓
[125]	✓	✓	✓	-	-	-	-	✓	✓
[67]	✓	-	-	✓	✓	-	-	✓	✓
[81]	✓	-	-	-	✓	-	-	-	✓
[214]	✓	-	-	✓	-	-	-	-	✓
[182]	✓	-	-	✓	-	-	-	✓	-
[80]	✓	-	-	✓	-	-	-	✓	✓
[76]	✓	-	-	-	-	-	✓	✓	✓
[74]	✓	-	✓	-	-	-	-	-	✓
[53]	✓	-	-	-	✓	-	-	-	✓
[126]	✓	-	-	✓	✓	-	-	✓	✓
[275]	✓	✓	-	✓	-	-	-	✓	✓
[254]	✓	✓	-	✓	-	-	-	✓	✓
[178]	✓	-	✓	-	-	-	-	✓	-
[278]	✓	✓	-	-	✓	-	-	✓	✓
[127]	✓	-	-	✓	✓	-	-	✓	✓
[274]	✓	-	✓	-	✓	-	-	✓	✓
[299]	✓	-	-	-	-	-	✓	✓	-
[27]	✓	✓	-	-	-	-	✓	✓	✓
[230]	✓	✓	✓	-	✓	-	-	✓	✓
[17]	✓	-	-	✓	-	✓	-	✓	✓
[228]	✓	✓	-	✓	-	✓	-	-	✓

evaluation, are presented.

Similarly, [26] broadcasts the first Interest and when a Data packet returns downstream, by bread-crumbs, the model creates dynamic unicast paths, by setting corresponding FIB entries, that will be further used to retrieve subsequent requests. The model is designed for MANET. Authors in [214] adopt the same idea but for VANET and, instead of using CCN Interest and Data for the discovery process, the consumer broadcasts a special packet (REQ), simpler than the Interest, in order to locate the content and then create unicast paths for subsequent packets. The response to the REQ is another special packet, REPLAY, more lighter than CCN Data, which is used to create the unicast paths. Based on nodes' dynamic parameters (e.g., distance, direction, traffic density) the proposal uses the included weight table and the neighbor table to calculate their dynamic score. The node with higher score is chosen as the next-hop.

Authors in [53] proposed an unicast-based routing protocol for efficient video sharing in vehicular networks. Having demonstrated in [273] that users with similar interests present higher probability to request similar videos, the proposal classifies the mobile users into multiple families (relying on the estimation of

mobility similarity and user preference similarity pattern) based on which the videos are shared.

Authors in [293] propose a hierarchical NDN-based routing scheme that uses BF [45] for content advertisement - Hierarchical BF Routing (HBFR). The authors assume that data in ICN can be categorized into: 1) popular public data services - where the services retrieve publicly shareable data; 2) popular private data services - related to popular data services that produce private data; and 3) unpopular data services, such as private messaging. Moreover, they consider that for each of these categories it may be necessary to choose an appropriate routing design category. Thus, they chose and designed their proposal for the popular data services (1 and 2). In HBFR, nodes in their corresponding clusters periodically summarize the content to create their content digests (a BF), which are then used to advertise (by flooding) the local content of the partition where they belong to. For rebroadcast avoidance, the study uses a purposely designed forwarding mechanism, based on timers for re-transmissions suppressing.

A reactive socially-aware routing protocol is proposed in [136]. The proposal is based on vehicle centrality metrics (leveraging the vehicles computing, caching and communicating capabilities) and confine the packet broadcast by routing it selectively towards information facilitator vehicles with the best vehicle centrality score [137, 138].

Authors in [263] propose a routing protocol that optimizes the routing path by a distance metric, in place of the hop-count, to estimate the link quality of communication nodes. Additionally, the proposal uses what the Authors dubbed as the incremental broadcast and adaptive broadcast strategy, which are based on the vehicle density (i.e., sparse, normal and dense) in the network.

Authors in [102] proposed a content routing based on Link Expiration Time (LET), which is used to calculate the expiration time of each interface listed on FIB and select from it the best routing path. This proposal does not describe much of the mechanism that take place to populate or maintain the FIB.

The study in [182] proposes a CCN-based framework for Inter Vehicular Communication (IVC), that first tackles the content naming (a scheme based on geographical and traffic information) in IVC, then routing (for relay vehicle selection) and caching search, to minimize the cache miss. In the proposed mechanism, Data packets are also forwarded by selecting relay nodes instead of using breadcrumbs.

The study by [80] explores caching for routing decision by allowing each vehicle in the network to perform flooding, requesting each other node to share the list of their cached content. In addition, the study refers to a supposed distance prediction algorithm, which calculate a next-hop close to the one forwarding the Data. The aforementioned method is not described in the study and it is unavailable. Authors in [230] present an idea for a context-aware routing and forwarding model, enhanced by mobility prediction and caching for routing decision. The proposal is based on the hypothesis that for better efficiency, routing should adapt to the specific environment or for each specific application type, because they present different characteristics among them.

A topic (i.e., road and safety, entertainment, public services) based clustering and routing mechanism is proposed in [76]. A periodic beacon (which includes the cluster's information topic and the expiration time) broadcast is the mechanism adopted by each Cluster Head (CH) to announce the existence of its cluster. One of the main aim of this proposal is to maintain and control the growth of the FIB. Also based

on clustering, [27] proposed a routing model where the single-hop clusters are formed according to the mobility of nodes (e.g., vehicles moving in the same region, on the same road and lane) using Hamming distance technique. A CH in each partition is responsible of performing data aggregation, intra-cluster communications and dissemination of content among clusters.

Authors in [299] combined the advantage of encounter-aware and clustering to propose a cluster and CCN-based routing protocol. The proposal introduces a new table, in place of FIB, that aggregates the encounter information. Based on this new table, CH are responsible on forwarding requests towards vehicles in the cluster or to other CH belonging to other clusters. The CH periodically updates and exchanges cluster tables in order to share the encounter information among them. Vehicles are clustered according to their trajectory. Also based on tracking the content locations and using last encounter information, a routing protocol is proposed in [292]. Vehicles keep exchanging their content list and locations whenever they encounter each others. Vehicles periodically advertise to one-hop neighbors a list containing a summary of contents in the node. When later a node receives an Interest with a name matching an entry on the list of content locations, geo-routing is used to forward the Interest. Other way, the Interest is flooded. For rebroadcast avoidance, the proposed scheme adopts a timer-based rebroadcast mechanism used to allow the broadcast when the timer elapses before the vehicle overhear a broadcast of the same packet. Also using timers for retransmission control, there is the proposal by [67], in which the Time To Live (TTL) is used to control the number of transmissions in each node.

Authors in [228] propose a cluster-based social vehicular routing protocol focused on avoiding broadcast storm when vehicular communication links breaks. Vehicles are clustered in communities. RSU concentrates information about all clusters and is responsible of electing new CH whenever the previous leaves the cluster. The work in [17] proposes a new routing algorithm based on the road traffic distribution. Vehicles opportunistically exchanges beacon messages in order to identify each other on the same road. For packet forwarding, the proposal uses, the exchanged information to build a map graph and compute a shortest path toward the content source. The well-traveled roads are chosen over the less-traveled ones. In [287], the Authors propose a routing scheme in which the degree of centrality and content interest are used as base for content routing. This proposal constructs community areas around the RSU coverage area. A Mobile Edge Server (MES), that deals with real-time content requests, is deployed in each community and is responsible of managing communications among vehicles and the RSU. A Fog Server, that deals with non-real-time content requests is deployed to manage the content distribution between two or more communities.

Authors in [74] proposed a routing protocol in which FIB entries are incrementally updated, instead of deleting and reconstructing them when the vehicle routes change due to its mobility. A new packet, dubbed resource packet, is added to the architecture and periodically broadcasted to announce new prefixes.

The CCN architecture is extended in [29] to include a so-called event packet, which can be seen as a modified unsolicited Data packet. The task of this packet is to disseminate push-based messages about emergency events or road traffic. In addition, FIB is modified in order to adapt it to be able to prefer and

choose some faces, depending on their characteristics (e.g., cost, latency requirement or bandwidth). The study in [81] extends the functionality of NDN for VANET by introducing a new packet - advertisement packet, that is send by the content source to announce its new content. In addition, FIB is modified to include the geographical location of the content source, from where the advertisement packet originated, instead of storing the next-hop information. Two content dissemination approaches are considered: 1) The advertised content, which refers to sending the advertisement packet; and 2) the on-demand content, where vehicles requests content not yet advertised.

Authors in [275] proposed a transmission protocol based on dynamic directional interface mapping, which in turn, is based on the node movement patterns. Vehicles are equipped with virtual interfaces which a remapped whenever driving directions of vehicles change. The remapping method is responsible for updating and maintaining the validity of FIB and PIT entries. One directional interface is selected for forwarding according to the position of the next-hop. All interfaces are used to broadcast when no corresponding FIB entry is matched.

A NDN-based framework for vehicle communications is proposed in [254]. In the proposal, vehicles and infrastructure cooperate to facilitate the communication process, and the RSU are responsible for caching popular content.

The study by [178] proposed a vehicular traffic information collection and routing protocol using Virtual Node (VN). A VN is a logical node maintained at a specific road intersection, it is responsible for collecting traffic information from all road segments adjacent to the intersection and share it with all vehicles passing by that intersection. When a virtual node moves from the intersection, another vehicle is selected and FIB, PIT and CS are transferred from the former VN to it, which then is assumed as a new VN. Routing relays packets preferentially via VN.

Authors in [274] proposed a hybrid routing protocol based on Name and ID. The protocol combines the routing based on names, from CCN architectures, and the ID-based routing, from IP-based architectures to address the mobility issue and the broadcast storm problem. In the proposed protocol, the nodes request data by CN. Then, having the content location, the ID-based routing is triggered. Using the corresponding position, the protocol computes the route towards the destination host. When the content provider is unknown, a flooding process takes place. In order to reduce the impact of broadcast storm, the flooding is controlled by counting the number of forwarded packets in each node. When this number reaches a specified threshold the node stops sending packets.

The study in [170] proposed a proactive routing which uses a Neighborhood Table (NT) as a complement to the FIB in order to maintain route entries towards the content sources. A periodic beacon is broadcasted to keep updated the NT and FIB.

Authors in [278] configured static FIB routes on NDN routers deployed in a hierarchical hyperbolic NDN-based backbone architecture, to achieve an efficient and scalable content retrieval from data centres. The model defines hyperbolic planes in the hierarchical architecture to offload the traffic to high-level routers, making static FIB entries able to forward packets.

3.8.2 Research Question (RQ2) - Caching

This section explains the process of data extraction related to RQ2.

For classification of caching scheme we use the combination of taxonomy and classification proposed in [139, 100, 140]. Table 14 presents the extracted data, based on the aforementioned parameters, for RQ2. The parameters used in the mentioned table are explained as follows: The type of scheme can be either:

- a) the content caching strategy, which is used to decide *where* to store *which* Data; and
- b) the content replacement policy, which is used to decide *which* content to evict when the CS is full and new content need to be stored.

The caching schemes can be grouped into:

- a) Popularity - when caching is performed based on the content requesting frequency;
- b) Cooperative - when the caching nodes cooperate to make a decision on where to store *which* content;
- c) Probabilistic - when the probability of content requesting is the main metric used to decide on caching; and
- d) Location based or mobility aware - when content is cached in pre-selected nodes, to attain higher cache hits (e.g: core/edge routers, BS, AP or RSU).

The caching mode can be either: a) reactive - when caching is triggered on consumer request; and b) proactive - when the content is pre-fetched, before it is requested.

3.8.2.1 Proactive schemes

The proposal in [179] proactively distributes and caches popular content in the selected and reserved CS, within the network. The caching mechanism is based on collaborative observation of locality in request frequency, which is assessed by means of a log record of overheard requests stored in each node, whenever it receives a location-dependent content request. Another proactive caching mechanism is proposed in [100]. The proposal proactively caches content at a predicted AP where a vehicle will connect to. To activate the mechanism, vehicles must request an initial INTEREST (which includes the current position/location, speed, INTEREST frequency, Data Object, RSU position, and the transmission range of the RSU) for a data object from the network. The provided information is used by the strategy to calculate the optimal chunk distribution onto the available RSU. Authors in [174] propose a proactive solution that formulates the optimization problem as an Integer Linear Programming (ILP) problem of optimally placing the content chunks. The impact of the following parameters on the retrieval probability:

Table 14: Extracted data from primary studies for RQ2 - Caching

Study	Pred.	Type of Solution		Caching scheme				Caching mode		Commun. mode	
		Strategy	Policy	Popularity	Cooperative	Probabilistic	Location	Reactive	Proactive	V2V	V2I
[97]	-	✓	-	-	-	-	-	✓	-	✓	✓
[205]	-	✓	-	-	✓	-	-	✓	-	✓	✓
[42]	-	✓	-	-	-	✓	✓	✓	-	✓	-
[157]	-	✓	-	✓	✓	-	-	-	✓	✓	✓
[73]	-	✓	-	-	-	✓	-	✓	-	✓	-
[100]	✓	✓	✓	-	✓	-	✓	-	✓	-	✓
[174]	-	✓	✓	-	-	✓	✓	-	✓	-	✓
[204]	-	✓	-	✓	-	-	-	-	✓	✓	✓
[265]	-	✓	✓	-	✓	-	-	-	✓	✓	✓
[235]	-	✓	-	-	-	-	-	-	✓	✓	✓
[272]	-	✓	-	-	✓	✓	-	✓	-	✓	✓
[1]	✓	✓	-	-	-	✓	✓	-	✓	-	✓
[180]	-	✓	✓	✓	-	✓	✓	✓	-	✓	-
[307]	-	-	✓	✓	✓	✓	-	✓	-	✓	-
[108]	-	✓	-	✓	-	✓	-	✓	-	✓	✓
[82]	-	✓	-	✓	-	✓	-	✓	-	-	✓
[84]	-	✓	-	✓	✓	-	-	✓	-	✓	-
[142]	✓	✓	-	-	-	-	✓	-	✓	-	✓
[164]	-	✓	-	✓	✓	✓	✓	✓	-	-	✓
[179]	-	✓	-	✓	✓	-	✓	-	✓	✓	-
[190]	-	-	✓	✓	-	-	-	✓	-	-	✓
[280]	✓	✓	✓	✓	✓	✓	✓	-	✓	-	✓
[250]	-	✓	✓	✓	-	✓	-	✓	-	✓	-
[109]	✓	✓	-	✓	-	-	-	-	✓	-	✓
[98]	-	✓	-	✓	-	-	-	-	✓	✓	✓
[111]	-	✓	✓	✓	✓	-	-	✓	-	✓	✓
[106]	-	✓	-	✓	-	-	-	-	✓	✓	✓
[282]	✓	✓	✓	✓	✓	✓	✓	-	✓	✓	✓
[79]	-	✓	-	-	-	✓	-	✓	-	✓	-
[110]	✓	✓	✓	✓	✓	-	✓	-	✓	✓	✓
[285]	-	-	✓	✓	-	-	-	-	✓	-	✓
[141]	-	-	✓	✓	-	-	-	✓	-	✓	-
[193]	✓	✓	-	-	-	✓	✓	-	✓	-	✓
[266]	-	✓	✓	-	-	-	-	-	✓	✓	✓
[99]	-	✓	-	-	✓	-	-	-	✓	✓	✓
[255]	-	✓	-	✓	-	✓	-	✓	-	-	✓
[107]	-	✓	✓	✓	-	-	-	-	✓	✓	✓
[306]	✓	✓	-	✓	-	-	-	-	✓	✓	✓
[248]	-	✓	-	✓	-	-	-	-	✓	-	✓
[281]	✓	✓	-	✓	-	-	-	-	✓	✓	✓
[24]	-	✓	✓	✓	-	-	-	✓	-	✓	-
[183]	✓	✓	-	-	-	-	-	-	✓	✓	✓
[103]	-	✓	✓	✓	✓	-	-	-	✓	✓	✓
[287]	-	✓	✓	✓	-	-	-	✓	-	✓	✓
[22]	-	✓	✓	✓	-	-	-	✓	-	✓	✓
[101]	-	✓	✓	✓	✓	-	-	✓	-	✓	✓
[123]	-	✓	-	✓	-	-	-	✓	-	-	✓
[59]	-	✓	✓	✓	-	✓	-	✓	-	✓	✓

number of users in the system, the AP available bandwidth, the propagation latency, and the size of the available caches, is also evaluated.

Authors in [204] propose a proactive and popularity-aware caching strategy where the content popularity is calculated both at the whole content and at the chunk level. Another cooperative- and popularity-based caching scheme is proposed in [280]. The proposed solution caches popular contents at a set of mobile nodes presenting the longest sojourn time in a given hot spot region. The proposal is based

on mobility prediction. The vehicle's mobility history is used to train the prediction algorithm (based on Partial Matching) in predicting vehicle's probability of reaching a given hot region. The same authors proposed in [282] another cooperative caching scheme with mobility prediction, but this time based on social attributes. The scheme design is based on the observation that moving vehicles are liable to contact each other according to common interests, or social similarities of their drivers/users. Yet another collaborative caching strategy based on socialized relations is proposed by [157]. The solution is designed only for video streaming and its caching decision is based on the minimum vertex cover set theory in static networks. Authors in [265] encode video streams into different layers depending on the video quality. When requesting video, the consumers indicate the desirable quality, then the intermediate nodes cooperatively cache the video chunks according to the requested layers. Another proposal for video streaming is proposed by [103]. This proposal is based on a two layer hierarchical architecture (core layer and the edge layer), with the nodes in edge layer grouped into clusters. Only edge routers perform caching. In addition to the content popularity, the proposal includes the predicted rating score for content placement decision.

Authors in [109] considered the problem of caching services in VANET, looking at both the mobility and storage. The problem was modelled using Markov decision processes. A heuristic Q-learning solution together with a mobility prediction scheme is proposed. Based on the speed of a requester vehicle, authors in [183] calculate the downloadable amount of an intended content to be pre-cached at the next RSU, where the vehicle is heading to. The proposal by [193] proactively distributes content at the RSU according to the movement of the vehicles. The strategy calculates the content size needed by a vehicle, caches some chunks onto present RSU and based on transition probabilities derived by MM, it distributes the remaining content chunks on other RSU. Also mobility-aware is the proactive caching strategy proposed by [142]. On content request, the proposed solution calculates and selects the next number of chunks that will be fetched by this vehicle and caches them on the next RSU, as predicted by the mobility prediction algorithm, proposed in [308]. Another proactive and mobility prediction based scheme is proposed in [1]. This scheme calculates and locates the pre-fetching node that minimizes uncertainty. The uncertainty is quantified by entropy. The mobility prediction algorithm is based on Markov models and a set of prior mobility traces. Authors in [235] considers the existing AP alongside the road for caching. Additional control packets are used to disseminate individual interests containing information regarding users' trajectories. Also with mobility prediction, authors in [110] propose a cooperative and cluster-based caching scheme that aims at mitigating the impact of node mobility on the network performance. Prior to caching, the proposed solution performs the clustering of vehicles presenting similar mobility patterns. Then, it creates communication among clusters by enabling the RSU to cache popular Data to the CH. The Authors prior study, in [111], does not include mobility prediction. It includes a replacement strategy based on content popularity.

With focus on autonomous vehicles, the proposal in [306] predicts a user mobility (based on the velocity and position) and the user's preferences (based on user's demand and previous video popularity), then proactively caches the videos that are likely to get requested at the next fixed infrastructure (i.e., RSU

or BS). The proposal in [281] takes advantage of the past mobility history of vehicles to estimate the time gap from the current location to the new (predicted by HMM) location. It uses the time gap to estimate the sequences of video chunks needed around that predicted location, and proactively caches them on the existing RSU, to provide them to the vehicle as soon as they arrive at the predicted location. Without predicting the vehicle mobility, the proposal by [98] learns the data popularity and prefetches popular content at the network edge AP. The same authors propose in [99], a cloud-like caching-as-a-service infrastructure, where opportunistically vehicles exchange their cache contents whenever they encounter each other. This scenario is referred to as a potential virtual cache area where vehicles proactively store and retrieve contents.

In [106], the content store is divided into two categories (i.e., proactive caching storage and reactive caching storage). The Zipf distribution is used to calculate the content popularity, from which the most popular content is proactively cached into the first caching category. In [107], the same authors proactively cache popular content at the RSU for better content distribution, when vehicles in the vicinity of the RSU require it. The cache prediction mechanism (which uses the principle of the content locality) is rewarded whenever a proactively cached content is accessed.

Authors in [248] added a new table (named Address Table (AT), for storing the addresses of left, right, and front RSU) to ICN. All content requests are forwarded to the RSU, which are responsible for caching contents. When a requested content is not available, the RSU forwards the request further to the BS, which is designed to have a broader information about all the RSU in a given area. Each RSU forwarding back the requested content, caches the content to their next left, right, and front RSU.

A caching strategy that uses machine learning to predict the requests for Interest and build a decision tree is proposed in [266]. The strategy works following the three steps: a) sample extraction; b) feature calculation; c) construction of decision tree and the pruning optimization. This strategy and an alternate replacement policy (based on popularity and request cost) are only deployed on the RSU.

3.8.2.2 Reactive schemes

In [205], a cooperative caching strategy based on social cooperation schemes is proposed. The cooperative scheme is grouped into: 1) partner-assisted - composed of vehicles within one-hop range and moving in the same lane with the consumer or producer vehicle; and 2) courier-assisted - composed by vehicles within the communication range of the consumer or producer, but circulating in adjacent lanes.

In [180] the caching decision is recommended by the content provider. The recommendation is based on the actual cache distance and frequency of the content requests. In [84], the decision on which content chunks to cache is based on content: a) popularity; b) vehicle connectivity; and c) the vehicle caching status, which is obtained by other vehicles with the assistance of a BS. In the aforementioned proposal, vehicles select and only cache some chunks of the overheard content. When a vehicle overhears a transmitted content, the algorithm proposed in [307] decides whether to cache the content or not. This decision is determined by assessing the content similarity which indicates whether the vehicle drivers

have similar hobbies. The authors also proposed a popularity-based prediction replacement mechanism, designed to reduce the overhead of cache replacement by predicting and evicting the content with the lower popularity whenever necessary.

Another popularity based scheme is proposed by [287]. This proposal divides the CS into two, i.e., 80% of CS for popular contents and the remaining 20% for the non-popular contents. A table, Content Popularity Monitor Table (CPMT), is included for monitoring the content popularity. CPMT is updated by abstracting the content name and type of Interest instead of Data. Also based on content popularity, [250] propose a caching strategy whose caching decision is based on the requirements of different types of applications and metrics, such as the vehicle link stability, content popularity, and CS occupancy. The first step performed by the strategy is to identify the application type based on the application name in the Data packet. Then, the decision is triggered and computed considering the identified application type.

Besides the popularity, [24] includes the content Residual Lifetime (RL) for caching decision. RL is included in order to limit the number of packet replacement due to lifetime expiration. In [22], the same Authors extend the previous work, adding the availability of the content in the neighborhood, to improve content diversity. Instead of using availability for content diversity, [101] distributes content evenly, resorting to the differences between the popularity of different types of content. This study introduces the node value (determined according to the connectivity, betweenness centrality, and eigenvector centrality) for selecting cache content and cache location.

A content location-aware and probability-based caching scheme is proposed in [164]. When a content is overheard, the proposed scheme calculates the content relative popularity and the cache interval, both metrics used to estimate the caching probability. For content popularity, different weights are used and larger weights are assigned to requests coming from nodes closer to the current RSU.

Authors in [42] propose three heuristic strategies (i.e., random caching, density aware, and forwarder-centered strategy) to reduce the cached copies produced by the NDN default strategy. For caching, the scheme chooses a sample of vehicles among all capable of caching, instead of allowing all vehicles to cache. The solution focuses on globally reducing the number of individual content copies instead of the cached content of a particular vehicle. The authors in [59], divided the IoV applications in categories (i.e., emergency safety, traffic efficiency and infotainment message), then designed a caching decision strategy tailored to the specific characteristics of each of these categories. For the specific case of the emergency safety messages, given that NDN uses pull-based mechanism for data dissemination, vehicle in the proposed solution periodically sends a series of interest packets requesting the traffic safety events.

The proposal by [73] focuses on probabilistically deciding whether to cache content or not, on intermediate nodes. This decision is evaluated in order to solve the problem of cache redundancy. Three factors are considered for this evaluation: 1) preference of vehicles; 2) importance of the vehicle, assessed by measuring the centrality of the vehicle in the ego network (local network formed by a node and its one-hop neighbors); and 3) relative movement of the vehicles.

In [97], the authors present a description and a performance assessment of their previous study [96], where they proposed a NDN-based VANET architecture. Considering the characteristics of VANET

such as the ability to provide continuous power supply and high computational capability (storage), and leveraging the wireless broadcast nature, the proposed architecture, [96], presents a caching scheme where instead of only caching content corresponding to the issued Interest, the design allows every node in the broadcast range to accept and store all overheard content, regardless of whether it has a matching PIT entry or whether it needs the received content.

Authors in [82] propose a Software Defined Networking (SDN)-based caching scheme where the RSU controller (instructed by the SDN controller) set chosen vehicles in the network, as influential nodes [36]. When the number of requests for a particular content reaches a previously defined popularity threshold (fixed by the SDN controller), the RSU and the content provider sends to these vehicles the popular contents to be stored. Designed for video streaming, [272] proposed a cooperative caching mechanism that is based on Information centric cloud. The Information centric cloud calculates the available load space for the nearby nodes of the requesting vehicle, to determine the proper cooperative caching nodes. The selected caching nodes are assigned the probability of caching, which increases with the available load space. The study in [123] proposes NDN-based networking architecture for IoV. The architecture, which is based on Real-time Onsite Operations Facilitation (ROOF) [165] is composed of three levels (i.e., a Cloud level, a Fog level, and a ROOF level). To facilitate content caching, the Authors propose a machine-learning-based data caching mechanism, used to predict the content popularity and store popular content in the ROOF level (i.e., in the RSU) and the less popular content in the Fog level.

3.8.2.3 Replacement schemes

The proposal in [285] incorporates the future content popularity for caching replacement decision. Metrics such as the characteristic of the received interests, the verified request ratio and frequency, and the content priority are used to access the content popularity. The content popularity is predicted by a HMM based algorithm. The received content is stored in CS by a descending order of the predicted popularity, and similar to [307], whenever necessary the content with lower popularity is (automatically) evicted.

A caching replacement policy based on multiple metrics is proposed in [190]. The scheme considers three factors for replacement decision: 1) the content popularity; 2) the freshness; and 3) the distance between the location of two specific nodes. When a replacement is required, the proposal uses the aforementioned factors to calculate the candidacy score for each content in CS. The three factors are all mandatory to calculate the candidacy. The average weight of each of these parameters is computed and the content with the lower weight is the candidate for replacement. Another caching policy, aiming at selecting and caching the highest popularity and evicts the lowest popular content is proposed by [141]. The proposal is based on: 1) content classification and cache splitting into sub-cache stores, according to traffic classification; and 2) caching and replacement whenever necessary, based on content popularity where each sub-cache store is allowed to only cache Data related to its class.

3.8.3 Research Question (RQ3) - Security, Privacy and Trust

This section explains the process of data extraction related to RQ3 - Security, privacy and trust.

Security and privacy issues posed by the underlying IP protocol stack in VANET [181, 224, 195] are naturally resolved by the core characteristics of NDN. However, new security and privacy challenges, specific for the underlying NDN architecture [57] still need to be tackled. Several vulnerabilities that can be exploited by attackers in NDN-based VANET include: a) Interest Flooding Attack (IFA); b) Cache Pollution/Poisoning Attack (CPA); c) False Information Injection Attack (FIIA); and d) Privacy Violation Attack (PVA) (e.g. vehicle tracking) by means of the content names [229, 90, 143, 215].

Although some security and privacy issues related to IP-based VANET are common to the NDN-based VANET, we only focus our attention on the security and privacy issues related to NDN-based VANET. Besides, we only survey solutions for NDN-based VANET not for the general NDN architecture (which can be found in studies such as [5, 187, 298, 39, 283]). The reasoning behind this exclusion is that we found several studies/surveys already treating the issues for the IP-based VANET that are common in NDN-based VANET.

Table 15 presents the extracted data for the proposed solutions for security and privacy in NDN-based VANET, and the synthesis of the studies is presented next. The *collaboration scope* field in the table indicates whether the solution requires collaboration among vehicles or if the nodes independently perform the countermeasure.

Table 15: Extracted data from primary studies for RQ3 - Security

Study	FIIA	IFA	CPA	PVA	Mitigation			Collaboration scope	
					Prevention	Detection	Recovery	Individual	Collaborative
[64]	✓	-	-	-	✓	-	-	✓	-
[63]	✓	-	-	-	✓	-	-	✓	-
[304]	-	✓	-	-	-	✓	-	-	✓
[143]	-	-	✓	-	✓	-	-	-	✓
[158]	-	-	-	✓	✓	-	-	-	✓
[217]	-	✓	-	-	✓	✓	-	-	✓
[153]	-	-	✓	-	-	✓	✓	-	✓
[309]	-	-	✓	-	✓	✓	✓	✓	-
[168]	✓	-	-	-	-	✓	-	-	✓
[284]	-	-	✓	-	✓	✓	-	-	✓
[206]	-	✓	-	-	✓	✓	-	-	✓
[3]	-	✓	-	-	✓	✓	✓	-	✓

Aiming at tackling false information injection and vehicle tracking threats, [64] proposed a four-level hierarchical trust model (which includes autonomous-vehicle organizations, manufacturers, vehicles, and data) and a naming scheme for vehicular authentication. To avoid vehicle tracking, the authors propose a pseudonym scheme whose responsibility is to anonymize vehicle names to protect vehicle identity. In [63], the same authors extend the previous work [64], to include an anonymous pseudonym-renewal scheme that combines self-insurance and third-party issuance, and pseudonymous authentication scheme which uses public-key cryptography.

The work in [168] also aims at detecting the dissemination of false information. Precisely the safety information. The authors propose the use of a set of rules, verified from the information provided by the vehicle sensors. The defined rules are the following: a) the vehicle should be on the move before the reported event; b) after the event, the vehicle should have reduced its speed; c) after the reported event, the vehicle should have its speed reduced by a certain value; d) the heart rate of the driver should have increased after the event; and e) the vehicle airbags should have been activated after the event.

To mitigate the CPA in the CS, the authors in [143] propose a reputation-based blockchain mechanism, in which newly connected nodes are assigned a reputation-value to indicate the degree of their reputation as cache stores. The reputation is increased whenever a consumer receives a valid content from the node CS. The trust is then based on the actual reputation value, the highest the value, the more trustable a specific node is. Another blockchain-based proposal, but for cache poisoning, is presented in [153]. The proposed mechanism is based on a three-stage scheme to detect all categories (i.e, corrupted content, unauthentic content, and fake content) of the content poisoning attacks: a) retrieval stage - for untrue poisoned contents identification, and elimination; b) recovery stage - for corrupted contents, that can not be identified by intermediate nodes due to the lacking of a signature verification; and c) feedback stage - for fake contents and potentially malicious content publishers identification. The work by [309] also aims at resolving the cache pollution problem. This work proposes a cache partition mechanism in which the CS is divided into two parts, and the content is stored in each part, depending on its popularity. A popularity monitoring algorithm is used to monitor the changes of the content popularity. The content with abnormal popularity is analyzed and accordingly marked whether it is an attack or not. A game theory-based mechanism to mitigate cache pollution is proposed in [284]. In this proposal, the RSU is responsible for gathering individual lists of suspicious vehicles, from the mobile nodes and generate the global black lists recording the attackers.

To mitigate the IFA, [217] adds a new field (`consumer_id`) in the Interest packet, and a new table (restricted id table), which is used to list the banned vehicles in the network. A threshold is fixed and a number of requests for a specific prefix is counted and compared to the threshold. When the counted requests reach the threshold, the requester vehicle is marked as suspicious and its id is broadcasted and included in the restricted id table. Considering that an attacker can bypass the proposed solution by requesting contents using a prefix distribution that for a specific name prefix does not exceed the fixed threshold, and therefore be undetectable, [206] propose the use of per-vehicle coarse granularity instead of per-prefix granularity, to overcome the mentioned shortcoming. Additionally, the proposal uses RSA-based cryptographic certificates in order to ensure non-repudiation, which is necessary for vehicle granularity.

A prevention and mitigation scheme for IFA is proposed in [3]. The mechanism monitor the incoming Interest per-flow instead of monitoring per-interface. The proposed per-flow monitor is composed of four components: a) a prefix analyser - to validate the naming scheme for the incoming Interest; b) the priority checker - which assign priority to the incoming Interest flow; c) suspicious list - the incoming flow that has low priority after some iteration is considered suspicious and moved to this list; and d) a scheduler -

responsible for sorting the flow of the incoming Interest prefixes according to its assigned priority. A node reputation value-based scheme for detecting a malicious node (responsible for IFA) is proposed by [304]. Nodes calculate their neighbors reputation value by direct and indirect evaluation. Then, by means of Markov chain and reputation history, they predict the current node state (in terms of reputation). These metrics are then used to determine whether to accept or discard the incoming Interest packet.

The name-based content identification (i.e., the hierarchical namespace with readable names), routing, and the security mechanism in which content providers cryptographically sign each Data packet, can be exploited to identify the intervening nodes. This can be done by correlating the Data requests and the Data names in the Interest packets. This situation can facilitate the PVA. To resolve this kind of issue, [158] presents an identity privacy protection strategy, which is based on ring signature [310] scheme. The ring signatures are used to hide the correlation between the signature and the data provider. In addition, the scheme uses the anonymous proxy to protect the real identity of the data requester.

3.9 Data analysis and Results Interpretation

This section presents the analysis of the extracted data and their interpretation.

3.9.1 Research Question (RQ1) - Routing

Although studies such as [97] consider that the main characteristics of VANET - high node mobility, dynamic topology, and intermittent connectivity - make it difficult or even infeasible to run a routing protocol to populate and maintain FIB, several efforts are being made to overcome such difficulties and develop routing solutions for NDN-based VANET. The main objectives of these solutions are: a) the elimination of the broadcast storm problem posed by the flooding approaches; b) the optimization of the resource usage; and c) the improvement of data retrievability.

In this work, for 9077 studies found, only 34 were selected for data extraction based on the inclusion/exclusion criteria. As presented in Table 13, the majority of the selected studies are designed for V2V communication model (See Figure 10).

The rationale behind this choice is that, although desirable to provide autonomy to VANET, giving it the capacity to perform all the needed actions to communicate without infrastructure assistance, the development of routing protocols for V2I communication is much necessary if one considers, for example, the need to leverage the SDN and Fog computing or the benefits of the RSU for caching in urban environments. Furthermore, the difference between the rural and the urban environments in terms of density, average speed of vehicles, and the existence of infrastructure to assist VANET may require the development of adaptable routing protocols, performing both communication models - V2V or V2I.

As proposed by [122, 78, 293], the operation mode of routing in VANET is based on: 1) topology - which in turn is divided in proactive and reactive; 2) geographical or position-based; 3) clustering; and 4) opportunistic encounters. The selected studies show that although some proposals are hybrid (i.e.,

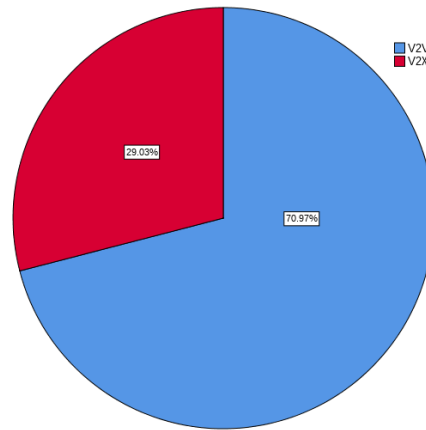


Figure 10: Communication model for the proposed routing schemes

classified into more than one operation mode, leveraging the synergies from these models), the majority of the proposed solutions are topology-based (See Figure 11).

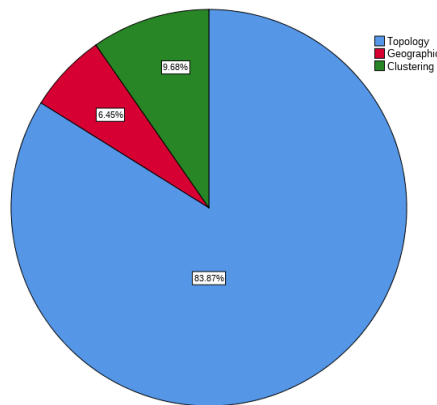


Figure 11: Operation mode of the proposed routing schemes. The topology-based mode is predominant.

Considering the VANET fast-changing topology and the fact that typical vehicular applications are geography- and time-dependent, and normally targeting vehicles within one specific region [197], it may be beneficial to combine the synergies of a hybrid mode of operation and a deep route/mobility prediction, mainly for the content provider. Although [277] proposes a scheme designed to route data packets considering the next location of a moving node, the process of acquiring the next locations is not explained. The aforementioned study only refers that the next location is sent within the Interest packet. If the next location is provided by the moving node, a way of updating this information is important. Furthermore, the presented solution only deals with the consumer mobility. A solution with mobility prediction, in which a node not only indicates the next location but, could also predict and indicate the mobility route and inform the network in case of deviation from the planned route could be considered. The proposal presented in [230], aims at tackling the aforementioned issue.

Topology-based routing schemes are divided into two groups: 1) one in which the routes to the content provider are frequently updated (proactively); and 2) the other where routes are determined on demand, whenever a new content request is received (reactive approach) [293]. Each of these approaches has

its own challenges. Frequently updating routing paths, in a highly dynamic and intermittent connectivity environment, can result in an excessive communication overhead.

For the topology-based proposals, 11 out of 28 studies propose a proactive topology-based solution. The choice for proactive routing is justified by the better performance of this approach, in terms of latency, as indicated by these studies. The overhead posed by this approach, however, is not negligible. Thus, additional work need to be developed to solve the excessive communication overhead. The majority of the studies propose reactive routing to avoid the aforementioned issue.

Six routing proposals are designed for location-independent applications, and only 5 are designed for the location-dependent application. The distinction between the two groups is important in the sense that, the location-dependent application are usually related to the emergency and safety applications, which only makes sense on a reduced area of interest, and share short-lived contents. The dissemination of this kind of content requires short delays, thus, broadcasting instead of routing them is the better solution. The work by [29] propose a mechanism to disseminate push-based messages, and modifies FIB in order to adapt it to be able to prefer and choose some faces, depending on their characteristics(e.g., cost, latency requirement or bandwidth).

None of the selected studies consider the incorporation of caching mechanisms in the routing design. To further mitigate the latency and the provision of content with the content provider mobility, the in-network caching may present a better solution. Considering parameters such as the type of information and mobility (of content provider or consumer), and the type of the network (rural or urban, and density of the RSU), a context-aware routing scheme able to dynamically adjust to network context could also be beneficial for VANET.

Table 16 presents the summary of the results to complement the data presented in Table 13.

Table 16: Summary of results for RQ1 - Routing

Study	Main features	Main limitations
[29]	FIB modified to be able to favor some faces depending on their characteristics (bandwidth, cost, etc.). Introduces a push-based mechanism for emergency- or safety-related content.	Routes in FIB may become obsolete as they are not frequently updated. Only for urban environments.
[292]	Uses exchanged last encounter information to keep updated the geolocation of content sources. Reduces flooding overhead.	Only deals with location location-independent content. Last encounter information may become obsolete, with less node encounters. Does not take in consideration the content source mobility. Only for urban environments.
[293]	Resorts to BF as a content digest to announce only the data service prefixes from the selected categories. Uses a timer-based mechanism to reduce the impact of the advertisement flooding. For mobility-awareness, the BF are periodically refreshed to reflect the topology change.	Classifies Data in three categories and only adapts for two categories. Assumes the availability of a network edge service used to classify the content. Only for urban environments.

Continued on next page

Table 16 – continued from previous page

Study	Main features	Main limitations
[277]	Proposes a push-based mechanism. Piggyback some sensing information (small sized) in the Interest packet. For mobility-awareness, the vehicle movement information is contained in the CN.	Only deals with location location-dependent content. Routes in FIB may become obsolete as they are not frequently updated.
[136]	Proposes a publish-subscribe ICN model. Avoids flooding by forwarding Interest to vehicles with high centrality. For mobility-awareness, path breaks are supported by beacon broadcasting.	Routes in FIB may become obsolete as they are not frequently updated. Only for urban environments.
[263]	Avoids beaconing by using estimated link cost as a distance metric instead of hop counts.	The mobility of the provider (<i>Responder</i>) is not taken in consideration. Only for urban environments.
[128]	Reduces message broadcasting by creating unicast paths from a requester to a content source. For mobility-awareness, path break supported by constant broadcast using the native NDN messages, instead of Beacon messages.	Uses broadcast MAC addresses to flood messages in the network. FIB entries may become obsolete. Only for urban environments.
[129]	Reduces message broadcasting by creating unicast paths from a requester to a content source.	Initial flooding necessary to populate FIB. FIB entries may get obsolete. Only for urban environments.
[102]	-	Use Face system without node identification. However, the wireless channel is broadcast by nature. If no corresponding FIB entry is found, the incoming Interest packet is dropped. Only for highways.
[170]	The neighborhood table, which acts as a FIB is periodically updated.	Periodic beacon broadcast to update the neighborhood table may increase network traffic. Only for highways.
[125]	Uses either vehicles (V2V) or RSU (V2I) for content retrieval. FIB is deleted each 10 seconds to keep it manageable and in comfortable size.	Continuous Beacon broadcasting to discover content sources, may increase network traffic. When no FIB entry is found the model delays the process until an entry is created. Only for urban environments.
[67]	Vehicles opportunistically obtain content source location from the received Data packets. Uses the link stability-based Interest Forwarding for Content Request (LISIC) [47] to control Interest flooding, when no corresponding FIB entry is found.	Does not consider mobility of the content provider. Recently obtained location information of a content provider may become obsolete before the content is requested.
[81]	For mobility-awareness, uses the store-and-forward (for consumer mobility) and floating content (for content source mobility) mechanisms, and periodic message retransmission.	-
[214]	Uses a neighborhood table and motion parameters (e.g., speed, direction, traffic density) to estimate the future location of a forwarding node.	Inclusion of nodes ID may re-introduce security and privacy concerns from the IP-based protocol. The process of discovering content sources (with new REQ REPLAY packets) then crating path may increase delay on retrieving contents.
[182]	Combines relay vehicle selection, packet forwarding and cache search. For mobility, performs periodic exchange of node position.	Only for urban environments.
[80]	Predicts the node trajectory. Vehicles maintain a table of neighbors' cache information.	Periodic Beacon broadcast with node position and cache information, among nodes, may increase network traffic. The inclusion of nodes ID may re-introduce security and privacy concerns known from the IP-based protocol.
[76]	Expiration time and the previous hop is used to avoid beacon rebroadcast. Prevents FIB from expanding too large.	Periodic beacon broadcast from CH to announce the cluster, and update FIB, may increase network traffic.
[74]	Incrementally updates the FIB to copy with the topology changes, instead of reconstructing it. Differently manages consumer and producer mobility.	Only designed for infotainment dissemination. Flooding a <i>hello</i> packet, alongside the <i>Resource</i> packet, increases network traffic. Only for urban environments.
[53]	Reduces bandwidth consumption and searching delay by using the unicast addressing mode.	Gathering users' behaviors may introduce privacy concerns. Only for urban environments and video fetching.
[126]	Uses directional antennas to direct traffic, and unicast communications to reduce network resource usage. A content-based forwarding, using timers, is proposed to support vehicle mobility.	FIB entries holding geographical coordinates may become obsolete given that they are not periodically updated.

Table 16 – continued from previous page

Study	Main features	Main limitations
[275]	An Interface remapping method is used to correct the directionless way of broadcasting, in wireless channel.	Only for urban environments.
[254]	-	Does not consider the mobility of nodes.
[178]	-	Only for urban environments.
[278]	Seamlessly switches hierarchical routing and hyperbolic routing for ensuring routing efficiency. FIB routes are static and use hyperbolic coordinates.	RSU is a key element for translating addresses. Thus, the proposal is not generalizable for environments without wireless static infrastructure. Querying the Location Name Service Server, may increase the delay for content retrieval. Only for urban environments.
[127]	Uses directional antennas to direct traffic, and unicast communications to reduce network resource usage. A Contention and time-based retransmission mechanism based on own proposed CBF.	The inclusion of nodes ID may re-introduce security and privacy concerns known from the IP-based protocol. Only for urban environments.
[274]	Leverages synergies from name-based, and ID-based routing for better mobility and flooding management. Resort to a BF-based CS structure for better content lookup. An algorithm to predict the link breaks and compute a new path in advance is proposed.	The use of IDs to identify vehicles may re-introduce security and privacy concerns known from the IP-based protocol. Only for urban environments.
[299]	Clustering is done according to the node mobility pattern. For mobility awareness, opportunistic exchange of node historical information, on node encounters is performed.	Nodes only make requests via CH, not among themselves. Traffic overhead at the CH can be high. Periodic beaconing for neighborhood discovering increases network traffic. The inclusion of nodes ID may re-introduce security and privacy concerns from the IP-based protocol.
[27]	Combines Dedicated Short Range Communication (DSRC) and Mobile Agent (MA) [152], and clustering for better routing decision. For mobility awareness, periodic information is exchanged among clusters.	Depends on availability of RSU, which are responsible for providing location codes to vehicles. Periodic information exchange among clusters increase network traffic.
[287]	A Fog network is used to facilitate the content sharing among different communities, thus improving content delivery rates.	RSU plays the role of communication medium, i.e., only V2I communications in the community area. Thus, a single point of failure. In each community, MES is a single point of failure. Only CH shares popular content via MES, with other communities.
[230]	Context-aware routing and forwarding. Application-aware caching. Routing and forwarding planes enhanced by mobility prediction.	The inclusion of nodes ID may re-introduce security and privacy concerns known from the IP-based protocol.
[17]	Beacon broadcast are only exchanged with encountered (opportunistic) vehicles along the road.	Map graph may rapidly become obsolete, increasing the risk of packet looping.
[228]	Performs a network load management.	Only for multimedia content dissemination. Community head and other vehicles cache information at a centralized entity, i.e, the RSU (a single point of failure). Community information table at RSU may rapidly become obsolete.

Possible Roadmap for Further Research In summary, a possible roadmap in terms of future work to further answer the RQ1 might be: 1) exploit in deep the content provider mobility prediction; 2) incorporate the mobility prediction in a context-aware V2X routing scheme; 3) design a model that can distinguish and prioritize content dissemination based on VANET application type, given that, for example, the safety-related applications are time sensitive, thus demanding highest priority; 4) exploit the incorporation of caching mechanisms to help mitigate the latency on content discover; and 5) further minimize the network overhead caused by the broadcast of infotainment-related applications.

3.9.2 Research Question (RQ2) - Caching

The critical challenge on improving Cache Hit Ratio (CHR) and access delay is the selection of an appropriate caching node within the network [285]. Standard NDN caching decision strategies are reactive-based, where the caching process starts with a content request, then, some copies of the content are cached somewhere along delivery path, according to the proposed caching strategy. For VANET, the conclusion from the selected studies indicates that proactive caching provides more efficient solution than reactive caching. As stated by [1], proactive caching can enable latency reduction on retrieving predictable content requests. Additionally, proactive caching can alleviate backhaul traffic and mitigate the latency caused by handovers.

Study by [197] presents key challenges for caching in VANET, such as: 1) scenarios with geographically disconnected segments; 2) environments involving both consumer, and provider mobility; and 3) development of caching techniques that consider spatial and temporal relevance, node mobility patterns, social relationships, and content popularity. The majority of the selected studies aim at solving the aforementioned challenges.

Forty eight studies, out of 5300, were selected for data extraction. As summarized in Table 14 and particularly shown in Figure 12, the majority (52.08%) propose a proactive mechanism for data caching. For both modes (proactive and reactive), the studies leverage the advantages provided by various caching schemes (e.g., popularity, cooperative, probabilistic, and location-based), for better cache management.

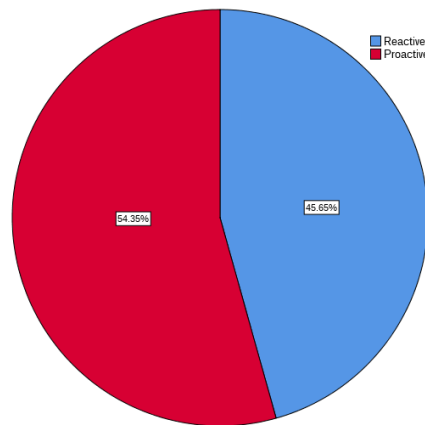


Figure 12: Distribution of the selected studies in terms of the proposed caching mode.

Three of the selected studies ([190, 285, 141]) propose only a caching replacement mechanism.

Eleven studies, out of the 25 proactive-based caching solutions rely on mobility prediction for path and node selection. In environments with high mobility, such is the case of VANET, the mobility prediction is challenging, however, fundamental for appropriate selection of next nodes for caching. Besides, considering the difference of characteristics between the location-dependent content (which is ephemeral and generally delay sensitive) and the location-independent content (which is long-lived and delay tolerant), an appropriate mechanism to differentiate and treat accordingly these groups of information is essential. All

proposals based on the mobility prediction ([235, 1, 142, 280, 282, 110, 193, 306, 281, 183]) rely on the existing infrastructure. Only 27.08% are pure V2V, see Figure 13.

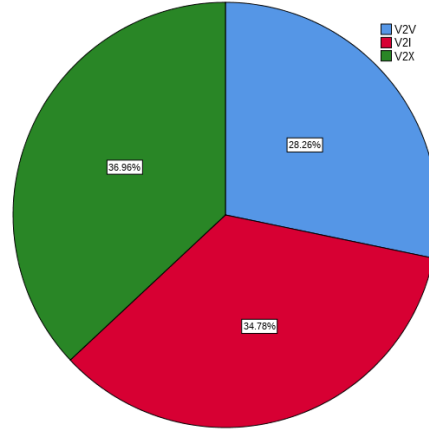


Figure 13: Distribution of the selected studies in terms of the communication model adopted by the proposal.

The majority of the selected studies relies on redundant caching at multiple nodes to reduce the effect of uncertainty presented by the proposed mobility prediction models. Studies in [1, 193], for instance, minimize this issue by quantifying it by entropy. However, the solution in [1] does not determine which content to fetch and how nodes are notified for pre-fetching.

The studies consider unlimited CS size, except [100, 174, 307, 280, 282, 110] which additionally propose a mechanism to manage the CS when it is full. Considering an unlimited caching size is clearly a limitation of the proposed solution.

Although all solutions are designed to solve the aforementioned challenges, they present some gaps that need to be tackled. For instance, the proposal in [174] is restrictive: nodes move along fixed routes and have a constant connection with the RSU. Chunks in each CS are considered as unchanging according to a caching policy. Study [100] makes a set of assumptions which are, in fact, limitations on the generalization of this proposal. The study considers only V2I communication with vehicles having constant speeds, and focuses on downloading only large files capable of being separated in smaller chunks for transmission. Furthermore, the study suggest the pre-fetching to all potential AP which may bring inefficiency to the caching system. In [193] each RSU is connected to a content server via backhaul link. This is a centralized approach. A single point of failure problem should be considered in this situation. The target in [205, 157, 204, 265, 272, 142, 103] is video streaming. Like the other studies, the contents are only pre-cached in infrastructure (i.e., RSU, BS, AP) not in other vehicles.

In [109] the mobility prediction model is developed by a deep learning model, capable of improving the mobility accuracy. But, due to the instability of VANET topology it should be well measured whether this advantage outweighs the introduction of additional complexity in the system, which may increase the latency. Study [110] uses the request frequency to classify the contents into most popular and least popular. The most popular content is stored in all CH. With this solution, within a given area the location

dependent content will be most frequently requested, thus becoming most popular and leading to cache this type of content for long time.

In [285] the social similarity is used to exploit the social relationship among nodes, bridging centrality to select important nodes for information dissemination. The authors justify their choice stating that a vehicle driving towards an area with sparse vehicles is not a good option to be selected for caching, because it may encounter fewer nodes. But, if we consider, for example, the need to connect geographically disconnected areas (using "data mules") or provide safety information to vehicles possibly heading to an affected area, such a vehicle may be of importance. In [280], the prediction and the key factor for choosing a vehicle as caching node is its past trajectories, and the long sojourn in a hot zone. As stated before, a vehicle without these requirements may still be of a importance to disseminate location-dependent contents.

Only [179] is designed for location-dependent content, where the information about the location is provided in the Interest/Data packet. This solution, however, does not predict the node mobility but it is stated in conclusion that increasing cooperation and cache coordination based on mobility patterns and group-forming can further increase the caching efficiency of this solution.

Cooperative schemes such as the proposed in [157, 265, 272, 282, 103], require sharing information among existing nodes. This additional signaling increases the network traffic and can not be negligible when measuring the overall network traffic.

Table 17 presents the summary of results, in terms of main features and limitations for each selected study, to complement the data presented in Table 14.

Table 17: Summary of results for RQ2 - Caching

Study	Main features	Main limitations
[97]	Takes advantage of the low vehicle restrictions in terms of storage, to cache all content and use moving vehicles as "data mules".	Only Urban environments. Considers unlimited CS size. Caches transient contents.
[205]	Improve QoE of multimedia streaming.	Only highway environments and multimedia streaming.
[42]	Three different heuristics are presented in order to avoid cache redundancy.	Only urban environments. Considers unlimited CS size.
[157]	Improve QoE of multimedia streaming.	Only highway environments and video streaming. Cooperation depends on the will of the neighboring nodes.
[73]	Factors such the preference of vehicles, importance of the vehicle and relative movement of the vehicles are taken in consideration for caching decision.	Only urban environments. Caches transient content. Users preferences and demands are extracted from the collected Interest, privacy concerns may arise.
[100]	Replacement policies: Least Recently Used (LRU), Least Frequently Used (LFU), First In First Out (FIFO). Mobility prediction using node position and velocity, INTEREST frequency.	Only motorway environments. Only deals with large files, which needs to be separated in chunks. Constant velocity. Signaling information required for calculating the list of chunks, which increases network traffic. RSU is a key resource and a single point of failure.

Continued on next page

Table 17 – continued from previous page

Study	Main features	Main limitations
[174]	Calculates the optimal CS size of the whole network in order to reach a desired retrieval probability.	Vehicles move along pre-defined paths. Caches only on RSU and AP. Chunks in the CS do not change over the time. Caches transient contents.
[204]	Independently switches the retrieval modes between V2V and V2I according the content popularity. Popularity metric enhanced by the reference frequency of the object (the chunk belongs to) and the reference frequency of the chunk.	Only urban highway environments and content streaming.
[265]	Replacements policy: LRU. Improve QoE by selecting content to cache based on the video specific bit rate.	Only urban environments and video streaming. Only caches on RSU.
[235]	Take advantage of the user trajectory information to select AP where to proactively cache desired content (i.e., predicts mobility).	All nodes promiscuously cache, but only RSU and AP cache proactively. Caches transient content.
[272]	Use Information centric cloud to calculate and decide in which node to cache.	Only urban environments and video streaming.
[1]	Predicts the node mobility (using 1 st -order MM). Use entropy to measure mobility prediction uncertainty and locate the best pre-fetch node.	Caches only on RSU.
[180]	Replacement policy: LFU. Besides popularity, the network knowledge (cache distance) is taken as advantage to increase cache diversity.	Only urban urban environments. Cache transient contents.
[307]	Replacement policy: PPCCR. Encourages the cache contribution of the private vehicles. Avoid redundancy by introducing popularity rating and hop number for selecting caching nodes.	Only urban environments. Caches transient contents.
[82]	Avoids network overloading with replicas of the same content by selecting influential vehicles as caching nodes.	Only highway environments. SDN controller, a centralized entity, being the owner of the caching logic can be a single point of failure.
[84]	Besides popularity, the caching status of neighboring vehicles is used for caching decision.	Only urban environments. Cached content do not change. BS being a key element responsible of sharing the caching status of the existing nodes can be a single point of failure.
[142]	Uses deep learning to predict the mobility of the nodes, to the next RSU.	Only highway environments and video streaming. Only caches on the RSU.
[164]	RSU perceive the surrounding cached-content locations by using the weighted recursive sum of the neighboring cache intervals.	Only urban environments. RSU are central entities responsible for caching and managing caching.
[179]	–	Only location-dependent contents. Beacon broadcast for social cooperation. Data packet have fixed and same size.
[280]	Own policy for cache replacement. Predicts the vehicle future location, based on trajectory history (using PPM).	Only urban environments. Caches transient contents. RSU are key elements responsible on recording trajectory records. RSU do not cache contents.
[250]	Replacement policy: LRU. Besides the popularity, considers other metrics (e.g., type of VANET application, cache occupancy, user's preference, and stability link of the vehicles) for caching decision.	Only urban environments.

Table 17 – continued from previous page

Study	Main features	Main limitations
[109]	Predicts vehicle mobility using Long-Short-Term Memory (LSTM), Deep learning.	Only urban environments and non-safety related applications. Only caches on RSU.
[98]	Uses content freshness and the number of hops for caching decision.	Only highway environments. Caches on edge routers (RSU, BS or AP). Designed for transient content (e.g., traffic updates, querying parking services).
[111]	Mobility prediction (mitigate mobility issues in V-NDN). Reduce redundancy by caching on selected nodes. Provides own cache replacement policy.	For urban environments. Only CH (which can not be accessible) and RSU (only available in some environments) cache contents. RSU selects CH for caching. Broadcasts beacon messages for sharing mobility information.
[106]	Solution combined with the cellular network (Long-Term Evolution (LTE) network infrastructures).	For urban environments. Requests are directed to the neighboring RSU which decides what to do further. Caches transient contents.
[110]	Take advantage of mobility prediction (using 2 nd -order MM) to cluster vehicles with similar mobility patterns. Provides own cache replacement policy.	For urban environments. Includes caching transient content. Signaling information required for node cooperation which increases network traffic. RSU are central entities responsible for selecting the caching CH, can be a single point of failure.
[282]	Predicts the vehicle future location (using HMM), based on vehicle behavior. Besides popularity, the social relationship is explored for better caching decision. Provides own cache replacement policy.	For urban environments. Caches transient contents. RSU are key elements responsible on recording trajectory records. Periodic beacons are broadcasted in order to share social attributes and determine hot zones.
[193]	Predicts vehicle mobility (using 1 st -order MM, and entropy for prediction accuracy).	For urban environments. Centralized content server. Caches only on RSU.
[266]	Uses decision tree prediction for caching decision. Provides own cache replacement policy on RSU and LRU in vehicles.	For urban environments. Caches transient contents.
[99]	Virtual cache area for proactive caching in infrastructure-less environments. Infrastructure-assisted cache-as-a-service. Use passing vehicles as data mules for caching.	For highway environments. Multi-hop communications not supported in all simulations. Caches transient contents. RSU beaconing for choosing potential caching node.
[255]	Resorts to Stackelberg game, as the cache allocation initializing procedure.	For urban environments. Vehicle beacon broadcast for sharing mobility information with one-hop neighbors may increase traffic. BS is a key element responsible of rewarding caching vehicles, a possible single point of failure.
[107]	Besides popularity, considers content locality. Resort to cloud servers as an additional caching resource. Includes own replacement mechanism (not described).	Caches only on the RSU. Caches transient contents.
[306]	Includes mobility prediction (mitigate mobility issues and predict user future demands). Uses 5G-ICN-based vehicular network.	For highway environments and video downloading. Caches only on BS and RSU.

Continued on next page

Table 17 – continued from previous page

Study	Main features	Main limitations
[248]	Provides a mechanism for the timely dissemination of safety-related messages (push-based).	For urban environments. Only caches on RSU and BS. Caches transient contents.
[281]	Predicts vehicle mobility using HMM, to mitigate mobility issues.	For urban environments and video downloading. Only RSU are responsible for caching and predicting vehicle mobility.
[24]	Besides popularity, includes RL for better caching decision. Uses RL-based eviction plus LRU policy.	–
[183]	Predicts vehicle mobility.	For urban environments. Only caches on RSU.
[103]	Besides the content popularity, the model includes the predicted rating score for content placement decision. Includes own cache replacement policy.	For urban environments and multimedia streaming. Only routers on the edge cache content.
[287]	Considering the friendship and in order to further improve sharing, nodes can suggest to cache non popular content into the reserved 20% of the CS. Includes own cache replacement policy.	For urban environments. The MES, a server deployed in each RSU and that manages communications in a community area, is a single source of failure regarding that specific community area (an area covered by a single RSU).
[22]	Besides the popularity, the caching decision is also based on the content residual lifetime and the perceived availability of the same content in the neighborhood. Includes own cache replacement policy.	Only the RSU is content provider.
[101]	Besides the popularity, the model introduces the node value concept for content and cache location selection. Includes own cache replacement policy.	No mobility awareness.
[123]	Caching is improved by leveraging Fog computing, to extend the CS capacity of the RSU.	Content is only routed through RSU. Only RSU and Fog are used to cache content.
[59]	Improve caching efficiency by differentiating IoV application content by classes. Includes a replacement mechanism. Besides the popularity, the caching mechanism includes spacial-temporal characteristics.	The sending frequency of emergency safety querying messages may affect the network traffic. RSU (a single point of failure) as a centralized entity used to determine the area of interest, for the traffic efficiency messages.

Possible Roadmap for Further Research The possible challenges to solve in order to further answer the RQ2 might be: 1) to develop a solutions with an improved prediction accuracy, designed to reduce cache redundancy on selection of caching nodes; 2) as the majority of selected studies do not distinguish between location-dependent and location-independent content, the development of caching solutions that can adapt to both type of information and application type, would be desirable; 3) develop a mobility-based caching scheme that could prefetch contents not only to infrastructures (RSU) but also to mobile nodes; and 4) design cache replacement mechanisms that can also include the type of content for caching replacement decision.

3.9.3 Research Question (RQ3) - Security, Privacy and Trust

This section presents the data analysis related to RQ3. As presented in Section 3.8.3, the main security and privacy attacks related to NDN-based VANET are: a) false information dissemination; b) Cache pollution/poisoning (these are actually different types of attacks [309], we just put them together given that we found scarce references for each, and are all related to the cache management).

Figure 14 shows that the majority (33.33%) of the selected studies deals with the FIIA and IFA, separately. Only one study deals with the privacy issue.

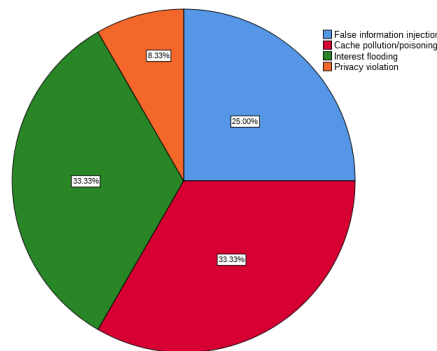


Figure 14: The surveyed main security and privacy attacks in NDN-based VANET.

In terms of collaboration scope, the majority of the proposed solutions impose that nodes work in collaboration among them. This type of solutions may result in inefficiency if one considers that VANET can be highly dense. Processing security algorithms in a highly dense and volatile environment can be challenging.

The study by [168] is specific for tackling the FIIA related to accident involving the vehicle that is disseminating the information, otherwise two of the defined rule (i.e., the vehicle airbags should have been activated after the event, and the rate of the driver should have increased after the event) could not be verified. This is a limitation concerning the generalization of this solution.

The proposal in [284], depends on the existence of RSU. Thus, this proposal can not be generalized to infrastructure-less environments. Another limitation is presented by the proposal in [158], which is concerned with protecting the identify of content requesters. The proposal does not generalize to protect the content source identify.

Possible Roadmap for Further Research Name-based routing provides natural countermeasures for Distributed Denial of Service (DDoS) attacks, by not providing the ID of the intervening nodes. The wireless channel is broadcast by nature, limiting the use of NDN Face system to identify the forwarding node. Several NDN-based routing for VANET, as presented in Section 3.8.1, propose the inclusion of nodes ID either to the Interest or Data packet. The inclusion of this node identifier could bring back the security and privacy vulnerabilities that exist in the IP-based VANET. Moreover, VANET scalability, partitioning and density, are some of the main challenges that can affect the design of a robust security,

privacy and trust solutions. Therefore, in summary, a possible roadmap in terms of future work might be: 1) designing a generalizable and scalable solution; 2) designing solutions capable of adapting to the density of the network; 3) given that trust and privacy are generally conflicting issues, the design of a mechanism that can equitably deal with both issues is desirable; and 4) advising new mechanisms to address the new security issues that arise by including the node ID in the NDN packets.

3.10 Improving VANETs using NDN

Implementation of VANET by NDN has been seen as a solution to overcome several challenges posed by the IP protocol. In fact, NDN characteristics (e.g., name based node identification and routing, in-network caching and the signing of each Data packet) provide natural solutions for several issues including: a) the node mobility; b) scalability; c) security and privacy; and d) content sharing model.

Although already presented for each RQ, we summarize here the lessons learned on improving VANET using NDN.

3.10.1 Research Question (RQ1) - Routing

In terms of routing, although opening a door for increased network overhead due to the frequent beacon broadcast to update the topology information, the proactive topology-based routing solutions are the most adopted. Taking advantage of all overheard packets to extract topology information, instead of broadcasting a specific packet, can be a better solution to avoid the network overhead.

If one considers, as explained in Section 2.2.2, that a solution is routing-based (belonging to the control plane) when a FIB management takes place, in contrast with flooding-based solution, which do not manage FIB, and simply broadcast the Interest into the network, then the reactive topology-based routing solution can be seen as belonging to the forwarding plane. The main issue presented by the latter solution is the delay on retrieving contents because the request is performed on-demand.

Besides the local-minimum issue, the location-based (or geographic-based) solutions are also prone to the delay issue presented by reactive topology-based solutions. To avoid the local-minimum issue, (only) one study propose geographical routing based on the (hierarchical) hyperbolic space. More research on this topic could be considered.

Although from the selected studies few hybrid solution (i.e., based on more than one mode of operation) were proposed, these have the potential for catching the synergies among the chosen schemes. Therefore, much research on the hybrid solution should be considered.

VANET communication depends on the wireless channel. The inclusion of node ID for node identification and relay selection is mandatory in this channel. However, security and privacy problems may arise by adopting this solution. Therefore, solutions proposing the inclusion of node ID require some mechanisms for tackling the aforementioned possible issues.

3.10.2 Research Question (RQ2) - Caching

In order to catch the synergies among the various caching schemes, the majority of proposed solutions adopt a hybrid scheme for caching decision. VANET application may either have short lifetime (e.g., the active safety or efficiency-related applications) or a relatively long lifetime (e.g., interactive-entertainment-related applications). Besides, the applications may be location dependent (e.g., the active safety or efficiency-related applications) or location independent (e.g., interactive-entertainment related applications). Therefore, additional metrics should be considered for deciding on which content to cache. One of the metric to consider is the lifetime of the received message. Actually, two of the selected studies do consider this metric. Caching content with short lifetime may result in an increased overhead on processing the CS, to frequently remove outdated content. More studies exploring this metric and considering the VANET application types (to consider the location dependency, for example) are desirable.

VANET should be capable of communicating without infrastructure (e.g., RSU, AP, BS). Although expensive, infrastructures are a normal reality on developed countries and are foreseen as being a key element to support VANET. It is easier to deal with the infrastructures, that are static, than with the mobile nodes. Having this in mind, several proposed solutions are developed to work with infrastructure. All proactive caching proposals explore the infrastructure (i.e., are either based on the V2I or V2X communication model). Proactive caching could also be considered in V2V communications, using the Floating Content (FC) concept [54, 131, 236], which is similar to the concept of VN proposed in [178] for content routing or virtual cache area, proposed in [99]. The mobility prediction, which the selected studies combined with the proactive caching, could also be explored with the aforementioned concepts to proactively cache contents in the infrastructure-less environments.

3.10.3 Research Question (RQ3) - Security, Privacy and Trust

The characteristics of VANET (e.g., sporadic connectivity, high speed of vehicles) impose specific challenges on developing security mechanisms. Therefore, security mechanisms should be designed with these aspects in mind.

Interaction among vehicles in VANET is a key aspect. Indeed, VANET is based on this interaction. Personal and private information (e.g., geographic coordinates, vehicle speed, or even vehicle identification in case of an accident investigation) is shared among vehicles. Generic security mechanisms capable of balancing privacy and trust are desirable.

Security mechanisms are required to take the delay constraints (crucial for active safety-based applications) into consideration. Therefore, they should be designed with low processing requirements and reduced messaging overhead. This is important mainly for the mechanisms that require collaboration among the nodes.

Node density in VANET can be sparse, normal or dense. Therefore, security mechanisms should be designed to scale proportionally to the node density and the network dimension.

3.11 Forwarding solutions

We highlight that this section was included later on this chapter. Initially the chapter were focused on routing, caching, privacy and security aspects. There are several solutions for content dissemination in NDN-based VANET. This work does not pretend to perform an exhaustive survey on the state-of-the-art of forwarding schemes, existing surveys on this matter include [140, 240, 8, 232] and the ones mentioned in Section 4.2.

Given the instability of routes in VANET, which is a direct result of the high mobility of nodes, network partitioning and short-lived connectivity, several studies (e.g., [251, 97, 239, 21]) argue that maintaining an updated FIB is computationally costly and not feasible, and that a flooding-based approach is the one that better suits VANET. Flooding performs well and is very robust in finding routes to the content providers on sparse networks [253, 23]; however, in a dense network, it brings a perverse problem that demands correction—the broadcast storm. In contention-based media, such as the case of wireless channels, flooding leads to a lot of conflicts in accessing the channel. The conflict results in packet loss, which in turn requires packet retransmissions. Packet retransmissions increase network traffic and worsen the channel's capability of disseminating packets, which results in deteriorated network performance.

Several solutions have been proposed for restricting uncontrolled flooding. These solutions include: 1) choosing a specific node, among others, which will forward the packet further; or 2) providing enough mobility information to the receiver node that it can use for deciding whether or not to forward the received packet further. For each aforementioned approaches, several other specific distinctions for better reducing the broadcast storm are proposed. For this, several parameters (e.g., geo-location, link stability or LET, and the distance to the provider or from the previous 1-hop sender) are usually employed. We highlight that this work strictly considers forwarding strategies as the solutions that do not directly manage FIB for routing/forwarding decisions. Studies that include FIB management for their forwarding decisions are considered to be routing protocols. This distinction is important and is extensively discussed in [232].

Based on the previously mentioned studies, including [240, 259], and taking into consideration the node that is responsible for forwarding the Interest, we categorize the strategies into two categories: 1) sender-oriented approaches; and 2) receiver-oriented approaches.

3.11.1 Sender-Oriented Approaches—Neighborhood-Aware

In this category, the node sending the Interest upstream is responsible for selecting the next-hop node that will forward the Interest further. Protocols in this class are also dubbed neighbor-aware solutions, that is, in order to be able to select a better relay for forwarding the Interest, nodes need to keep a list of surrounding nodes (i.e., neighbors) from which the next forwarder is chosen.

State-of-the-art protocols in this category [160, 258, 259, 154, 11, 40] resort to beacon broadcasting for discovering new neighbors and updating the positions of the already existing ones. Broadcasting an additional control packet into the network results in an increased transmission overhead. The frequency of

the beacon broadcast should be chosen wisely, taking into consideration that, although broadcasting with high frequency is better for updating the list of neighbors, it also results in increased traffic, which may result in a broadcast storm, and consequently deteriorated performance. This problem is exacerbated in dense network scenarios. Conversely, sending less frequent beacons is better for keeping the traffic low and the channel reserved for the actual content dissemination. This, however, results in an outdated list of neighbors, which in turn affects negatively the efficiency of the protocol. Therefore, a compromise between these two factors (i.e., transmission overhead and the timely updated list of neighbors) should drive the selection of the frequency for beacon broadcasting.

Instead of using beaconing for neighbor discovery, the proposal in [186] uses a pair of digital cameras for collecting the visual identification of vehicles in the range (line of sight) of the cameras. The need for a line of sight, and the complexity of processing the visual data (i.e., license plate, color, brand, and car type) and accurately distinguishing the neighbors may be challenging, mainly in a dense network. Another alternative to beacon broadcasting is the overhearing of the packets in the wireless channel and using them to build and update the list of neighbors. Moreover, as proposed in [232], incorporating packet overhearing and mobility prediction for updating the list of neighbors may avoid the need for beacon broadcasting and help to keep the transmission overhead low.

Protocols in this category are unicast by nature; as previously explained, they forward a packet to a specific selected node. In order to be able to identify a specific node, nodes should be unambiguously identifiable; therefore, usually, additional fields (i.e., MAC address of the node's Face) are added to the packets. Furthermore, the mobility information of the nodes also need to be added to the packets, either by including new fields in the Interest-Data packet or by piggybacking the information into NDN Link Protocol's headers. In sender-oriented approaches, the next-hop forwarder is selected considering parameters such as: a) their distance (mostly the Euclidean distance) from the 1-hop previous node; or b) their distance (Euclidean distance or the number of hops) to the content provider, if its location is known. The following are the representative state-of-the-art solutions into this category.

In the Geographical Opportunistic Forwarding Protocol (GOFPP) [160], a next-hop forwarder is chosen if it can satisfy the Interest itself (being a producer or holding the desired content in its CS), or if it can be the earliest to reach the Point-of-Interest (Pol) where the desired content can be fetched. So if the node does not hold the desired content in its CS, the minimum distance to the Pol is the parameter chosen for selecting the next-hop forwarder. If no vehicle meets the two previously mentioned conditions (i.e., Data in CS and existing node for calculating distance to Pol, or the current node with minimum distance to it compared to possible forwarders), the node holding the content does not forward it, and keeps it until a suitable node is located. For selecting next-hop forwarder for the Data, the nearest distance between the trajectories of the candidate next-hop and consumer before the Data's freshness expires is considered.

The Mobility-Predict-based Forwarding Strategy (MPFS) [258] uses two parameters for selecting the next-hop forwarder: a) the LET, and b) the distance along the road. The node with the highest LET and simultaneously farthest distance along the road, within the transmission range of the sender, is selected

as the next-hop. The solution presented in the aforementioned study only consider highways, and two next-hop forwarders are selected, one for each direction. Moreover, besides the mobility prediction, the solution still resorts to broadcasting beacon (every 2 s) for updating the NT. A similar solution, Predictive Forwarding Strategy (PRFS) [259], is presented by the same authors.

Similarly, two next-hops forwarders are selected in the Packet Forwarding strategy based on Optimal and Backup (PFOB) [154], one for each direction of the highway. This protocol uses distance, relative speed, link duration, signal strength, and node degree for selecting the next-hops. During the process of selecting a next-hop to forward the Interest upstream, simultaneously, a backup node is chosen for helping to forward the Data downstream. The chosen backup node is the one with the highest link stability. Beacons for updating NT may increase the network overhead.

In RobUst Forwarder Selection (RUFS) [11], nodes exchange, via beacon broadcast, a so-called recent satisfied list which includes satisfied Interests for specific CN, number of hops for the satisfied Interests, and the Interest Satisfaction Ratio (Interest Satisfaction Ratio (ISR)). The received information is used for updating a local list called the Neighbor Satisfaction List (NSL). A node from NSL that simultaneously presents the highest ISR and lowest number of hops to the content source, and that recently satisfied the requested content is selected as the next-hop forwarder. Beacons for updating NSL may increase the network overhead.

As highlighted before in this section, the maintenance of the NT is a crucial factor for the efficiency of the protocols presented in this section. The NT is invaluable if not properly updated. Using an outdated NT for selecting the next-hop forwarder can result in an erroneous selection of a next-hop (i.e., the node may already be out of the transmission range of the sender), leading to the loss of the forwarded packet.

3.11.2 Receiver-Oriented Approaches

In this category, instead of selecting the next-hop forwarder, the sender broadcasts the Interest and leave the responsibility of deciding whether or not to forward the Interest further to the nodes that received the packet. Nodes calculate their relative forwarding priority, and nodes with higher priority retransmit the received interest. Nodes with lower priority only retransmit the received Interest if they do not overhear a retransmission of the same packet. Protocols in this category are basically timer-based forwarding approaches. When a node receives an Interest, it sets up a deferring timer and awaits to overhear a retransmission of the same Interest from other nodes. When the node overhears the same Interest retransmitted by other nodes, it cancels its own waiting timer. Nodes with higher priorities have a shorter waiting timer. Nodes calculate their priorities taking into consideration parameters such as: a) their distance (mostly the Euclidean distance) from the 1-hop previous node, and their distance (Euclidean distance or the number of hops) to the content provider if its location is known; b) link stability; c) Interest satisfaction rate; and d) their geo-location and other centrality measures. Some of the state-of-the-art solutions in this category are presented next. Some of the protocols presented next calculate node priority based on more than one of the aforementioned criteria.

3.11.2.1 Position- and Distance-Based Approaches

Strategies in this category [262, 97, 156, 150, 149, 65, 290, 212, 15, 14, 294, 69, 256, 295] select the farthest nodes from the 1-hop previous sender as the next-hop forwarders. The distance to the content provider, if available, is also used to further refine the next-hop selection.

Aiming at tackling the broadcast storm and network partitioning problem, Rapid Traffic Information Dissemination (RTID) [262] relies on four timers (i.e., collision avoidance timer, pushing timer, NDN-layer retransmission timer, and application retransmission timer) to achieve the aforementioned objective. The “collision avoidance timer” is used for delaying transmission among neighbors and, thus, avoid collisions. The “pushing timer” is used for defining a next-hop forwarder, which must be the farthest one from the actual sender. The “NDN-layer retransmission timer” is used for re-broadcasting and reception acknowledgment, and the “application retransmission timer” is used for Interest re-expressing from the applications. V-NDN [97] uses only one timer for mitigating the broadcast storm. The waiting timer calculation is based on the locations of the 1-hop previous nodes and Data sources. A farthest neighbor from the previous 1-hop node that is concomitantly the closest node to the content provider has the highest priority for forwarding; thus, its waiting timer is the shortest. The proposed solution spreads the Interest in all directions.

The Context-Aware Vehicular NDN (CA-VNDN) [156] uses the same idea but restricts the Interest forwarding to a specific area. Furthermore, [156] includes an enhancement of Data breadcrumbs forwarding by unicasting the Data to its provider if its geo-location is present in PIT.

Another proposal similar to [97] is Density-Aware Delay-Tolerant (DADT) [150]. The proposed solution adds the capability of supporting network partitioning by storing every Interest packet in a so-called pending retransmission queue, from which the Interest is only retransmitted when a capable neighbor can be selected for transmission. The protocol broadcasts a beacon every second, which may increase the transmission overhead. Additionally, the proposal requires the node to know the producer’s geo-location, which may not be provided. Two timers (i.e., Forwarding-Priority timer and Collision-Avoidance timer) are used in their other work, dubbed Location-Based Deferred Broadcast (LBDB) [149]. The “Forwarding-Priority timer” is used to determine the priority and selection of the next-hop, and the “Collision-Avoidance timer” is used for preventing collision. Also, based on the distance from the 1-hop previous node and closeness to the content provider, Content Connectivity and Location-Aware Forwarding (CCLF) [65] also resort to a queue for storing Interests, which are then retransmitted when a neighbor is sensed. It sends periodic beacons for maintaining the NT when no packets are sensed for some time. A particular characteristic of this proposal that is different from [150] is that the neighborhood awareness is also used for suppressing Interest–Data, and thus efficiently manages the node’s resources.

Nodes in Opportunistic Interest Forwarding Protocol (OIFP) [290] only consider the distance between themselves and the 1-hop previous node to compute the priority and auto-select themselves as the next-hop forwarders. The closest node to the boundary of transmission range sets itself as the priority forwarder by setting its waiting timer lower. This proposal differentiates itself from the former by sending the Interest

to different directions. In [290], nodes send the Interest even when there are no neighbors around, which results in an inefficient usage of the node's resources. In [256], prior to sending the content request, a beacon broadcast, for provider discovery, takes place. Only when the location of the content provider is known is the contention timer set and the content requested.

Besides the distance from the 1-hop previous node, the Content Discovery Protocol (CDP) [212] also uses the so-called sweet spot (i.e., an area where the vehicles are best positioned to forward packets to a higher number of neighbors) to further refine the process of prioritizing the next-hop forwarder selection. Thus, a node receiving an Interest sets its priority high, and consequently the waiting time for retransmission is lower if its distance from the previous node is relatively higher, considering the transmission range of the sender, and the node itself is within the sweet spot. The Interest–Data flow-tracking-based forwarding scheme (IDTracS) [15] uses the distance from the 1-hop previous node alongside the degree of centrality (i.e., counted Interest and Data packets) to decide which node should forward the received Interest.

Differently from the study in [11], which uses ISR as one of the metrics for selecting the next-hop, [15] uses the counted Interest and Data packets independently. The farthest nodes from the previous sender that simultaneously present a higher degree of centrality have higher priority and thus a lower waiting timer. The same authors propose in [14] a protocol that creates suppression areas for controlling and suppressing duplicated Interest forwarding. In the defined area, timers are used to prioritize forwarding. The road topology is used to specify the areas which are created in different directions, leading to a forwarding strategy that forwards Interests to each road direction. Context-Aware Content-Naming (CACN) [294] and Decentralized Receiver-based Link Stability-aware Forwarding (DRLSF) [295] also propose the creation of forwarding zones, which are based on the angular information of the nodes on the road. It catalogs the created zones (i.e., red and white zones) regarding their capacity for forwarding packets with fewer unnecessary retransmissions.

3.11.2.2 Link-Stability-Aware Approaches

Strategies in this category [12, 237, 47, 48, 69] select nodes with relatively more stable links as the next-hop forwarders. The idea is to ensure that the returning path for the Data packet is still available when the content is sent back downstream.

In Distributed Interest Forwarder Selection (DIFS) [12], two forwarders are selected as next-hops, for each of the two highway directions. The next-hop forwarder auto selects itself among neighbors if it cumulatively has a long link connectivity duration, it is the farthest node, and it has a relatively lower speed compared to the previous 1-hop node. Two lists (i.e., neighbor list and Decision List) are used in each node for deciding on the forwarding eligibility. The DL holds the status information of all neighbors and each node uses it for ranking itself among them. Beacons are used to share the mobility information among neighbors, which may increase the network overhead. Moreover, the information in DL may become outdated and affect the eligibility decision and consequently the efficiency of the protocol. In Link

Stability-Based Interest Forwarding (LSIF) [237], when a node receives an Interest, it extracts the mobility information from the received Interest and estimates the link lifetime, i.e., the time the two nodes will be in their mutual communication range. It periodically predicts the node mobility and estimates the remaining lifetime of the link. If the estimated lifetime is above a defined threshold, the receiving node forwards the packet; otherwise, it drops it. The original sender node always sends the packet regardless of not having any node in its vicinity.

Similarly, aiming at reducing redundant transmissions of Interest, Link Stability-Based Interest Forwarding for Content request (LISIC) [47] creates stable paths for Data delivery by selecting more stable links for forwarding Interests. A waiting timer is computed based on the estimated duration of the link lifetime. Furthermore, [47] resorted to hop count for further limiting the propagation of Interests. The same authors proposed in Location-based and Information-Centric (LoICen) [48], a protocol that resort to the opportunistic sharing of CS content catalog among vehicles. A so-called Content Location Table (CLT) is used to store the location of nodes previously encountered. Interests are then forwarded towards the location where the vehicles holding the content are located. The aim of this proposal is also the avoidance of broadcast storm by redirecting the content for specific locations where a possible holder is located. Outdated content location may lead to packet loss and retransmissions. Without a possibility of updating or predicting the mobility of the nodes registered in CLT, information in this table may rapidly become outdated.

The proposal in E-Fuzzy [69] includes also the distance between nodes and the signal strength, and resorts to fuzzy logic with the aforesaid metrics to identify the best nodes for forwarding.

3.11.2.3 Geo-Location-Aware Approaches

Protocols in this class [201, 262] forward Interest to a specific region. Generally, the requested content is location dependent and only makes sense for that specific location. Any node, even outside the region in which the content is desirable, can forward the content should it possess the requested content in its CS. For location-independent content, the protocol is required to use additional measures to tackle the producer's mobility problem.

In GeoZone [201], Interests are forwarded using a geo-referenced naming scheme to request content from a desired POI or dissemination zone. When an Interest is issued, only nodes within the POI can flood it, and the flooding is limited to that given zone. A node receiving an Interest decides whether or not to forward it further based on its own geo-position. If within the dissemination zone, it can forward the Interest within the zone. For collision avoidance, the protocol adopts and uses three of the four timers proposed in [262].

3.11.2.4 Interest Satisfaction Rate-Based Approaches

In Neighborhood-Aware Interest Forwarding (NAIF) [291], the next-hop forwarder auto-selects itself considering its Data retrieval rate (i.e., ISR) and a forwarding rate (i.e., fraction of incoming Interests the node

will forward) for a specific CN, and the distance to the consumer. Nodes in the proposed protocol can reduce the overhead by sharing the workload among their neighbors. Forwarding statistics are updated by periodically probing the 1-hop neighbors. The probing process introduces a new packet into the network and may result in an increased transmission overhead. As mentioned before, differently from this study, the work by [15] uses the counted Interest and Data packets independently, instead of computing the ISR.

3.11.3 Comparison of the Selected State-of-the-Art Solutions

The following section summarizes and compares the related work on forwarding strategies. Table 18 shows in column “Forwarding”, the alternative solutions as presented in the previous sections. The options are sender-oriented, which are unicast-based solutions, and receiver-oriented, which are mainly broadcast-based forwarding schemes. In terms of new packets, the protocols that include new packets resort only to the beacon broadcasting.

Our proposal is designed to work without resorting to beacons. However, it alternatively processes beacons from the RSU whenever they are received. It means that the solution adapts itself for working with beacons should they be received. As shown in Section 6.3, the alternative without processing beacons presents better performance. The “N. St.” field indicates an additional structure besides the common ones from the native NDN (i.e., CS, FIB and PIT). The “N. Pack.” indicates whether or not additional packets, besides Interest and Data, are used by the protocol. Generally, a beacon or additional Interest carrying mobility information or a possible catalog of cached content is used for this matter.

CMAF have two distinct operational modes, one for V2V and another for V2I. In V2I mode, a beacon is broadcast by the RSU or vehicles; therefore, the field “N. Pack.” should be considered differently for each operational mode. The “Neighb.” field indicates that the protocol manages a NT and only sends Interests when the lists are not empty—we highlight that some protocols do not regularly update their neighborhood list. The “Other” field indicates the use of other criteria, such as relative speed for calculating, for instance, which vehicle will reach the Pol first (e.g., study by [160, 154]), node degree, or signal strength (e.g., in [154]), and suppression angle (e.g., in [156]).

CMAF chooses the next-hop forwarder somehow similar to the proposal by [150]: the sender chooses from the NT a node that is: a) farthest from the previous 1-hop node; and b) that is simultaneously the closest to the content provider, whenever this information is known. Moreover, the model uses hop count to further limit the Data packet dissemination. Differently from [10], the hop count is fixed based on the average number of neighbors for a previously predetermined time as presented in Section 4.3.3.2.

Table 18: Summary and comparison of representative state-of-the-art solutions.

Study	Forw.	Criteria for Next-Hop Selection					Comm.		Main Challenges/Limitations				
		Dist.	Link- stab.	Geo- locat.	Pack. (Rate)	Neighb. Other	M. Pred.	Scen.		N. Pack	N. St.	V2V	V2I
[97]	Receiver	-	-	✓	-	-	✓	Urban	-	-	✓	✓	Fixed HopLimit. Only location-dependent content.
[259]	Sender	✓	✓	-	-	✓	✓	Rural	✓	✓	✓	-	Additional overhead from beaconing for neighbor discovery. Geo-location provided but content source's location awareness not leveraged . NT is updated reactively and the listed neighbors in the NT may already be out of the transmission range of the sender.
[160]	Sender	✓	-	-	-	-	✓	Urban	✓	-	✓	-	Additional overhead from the opportunistic CS content digest announcements. Does not consider producer mobility.
[258]	Sender	✓	✓	-	-	✓	✓	Rural	✓	✓	✓	-	Additional overhead from beaconing for neighbor discovery. Mobility prediction only considers linear trajectory of vehicles.
[154]	Sender	✓	✓	-	-	✓	✓	Rural	✓	✓	✓	-	Additional overhead from beaconing for neighbor discovery. Geo-location provided but content source's location awareness not leveraged.
[11]	Sender	✓	-	-	✓	✓	-	Urban	✓	✓	✓	-	Additional overhead from beaconing for neighbor discovery. Geo-location provided but content source's location awareness not leveraged.
[262]	Receiver	✓	-	-	-	-	-	Rural	-	-	✓	✓	Vehicles with a constant speed.
[156]	Receiver	✓	-	-	-	✓	✓	Urban	✓	✓	✓	-	Additional overhead from beaconing for neighbor discovery.
[150]	Receiver	✓	-	-	-	✓	✓	Urban	✓	✓	✓	✓	Additional overhead from beaconing for neighbor discovery.
[149]	Receiver	✓	-	-	-	-	✓	Urban	-	✓	✓	✓	Producer mobility not considered.
[65]	Receiver	✓	-	-	✓	✓	-	Urban	✓	✓	✓	-	Additional overhead from beaconing for neighbor discovery.
[290]	Receiver	✓	-	-	-	-	-	Urban	-	-	✓	-	Geo-location provided but content source's location awareness not leveraged.
[212]	Receiver	✓	-	-	-	-	✓	Urban	-	-	✓	-	Geo-location provided but content source's location awareness not leveraged.
[15]	Receiver	✓	-	-	✓	-	✓	Urban	-	-	✓	-	Geo-location provided but content source's location awareness not leveraged.
[14]	Receiver	✓	-	-	-	-	✓	Urban	-	-	✓	-	Requires knowledge of different topological road structure. Geo-location provided but content source's location awareness not leveraged.
[294]	Receiver	✓	-	✓	-	-	✓	Rural	-	-	✓	✓	Considers an immovable producer. Only disseminates notification messages.
[12]	Receiver	✓	✓	-	-	✓	✓	Rural	✓	✓	✓	-	Beaconing for sharing mobility information may increase overhead.
[237]	Receiver	✓	-	-	✓	-	-	Urban/ Rural	-	✓	✓	-	Assumes nodes following a linear trajectory.
[47]	Receiver	-	✓	-	-	-	✓	Urban	-	✓	✓	-	A node with similar mobility pattern (i.e., velocity and direction) as the sender may not always be the better next-hop, and it may be better to keep the Interest with the current node. Low network density not considered.
[201]	Receiver	-	✓	-	-	-	-	Urban	-	-	✓	✓	Geo-location of the content must be known.
[291]	Receiver	✓	-	-	✓	-	-	Urban	✓	✓	✓	-	Additional overhead from beaconing for collecting forwarding statistics.
[48]	Receiver	-	✓	-	-	✓	-	Urban	-	✓	✓	-	Mobility of vehicles holding content in their CS is not considered. Outdated content location may lead to packet loss, and require retransmissions. Low network density not considered.
[295]	Receiver	✓	✓	-	-	-	-	Urban/ Rural	-	-	✓	-	
[186]	Sender	-	-	-	-	✓	-	Urban	-	-	✓	-	Assumes the node transmission range equal to the range of cameras in order to ensure the reachability of the line of sight.
[40]	Sender	-	-	-	-	✓	-	Urban	-	-	-	✓	NT is not updated.
[69]	Receiver	✓	✓	-	-	-	-	Rural	-	-	✓	-	
[256]	Receiver	✓	-	-	-	-	-	Urban	-	✓	✓	-	
CMAF	Sender	✓	-	-	-	-	✓	Urban/ Rural	✓	✓	✓	✓	Computation cost for proactively predicting vehicle's mobility to update the NT.

3.12 Summary

Resorting on a systematic literature review, this work proposed to investigate the existing solutions and the main open challenges on how to improve VANET using NDN. Three research question were defined and a total of 94 studies spanning from 2010 to 2021 were selected for data extraction.

This extracted data shows that studies within the selected topics are recent (from 2016 to 2021) and the majority of them (i.e., 51.06%) deal with the caching system.

Existing gaps extracted from the cumulative findings and the future work presented by the studies themselves were presented for each topic, highlighting the need for studying and better integrating the mobility prediction with routing and caching.

In summary the present SLR recommends:

1. The development of work aiming at improving the mobility prediction accuracy;
2. The development of a context-aware routing and caching scheme, considering the different characteristics of the network environment, such as the network density, the type of application (for location-dependent/independent content), the model of communication (V2V/V2I), and the integration of caching mechanisms in routing decisions with mobility prediction;
3. The development of a more generic security and privacy solution for better prevention, detection and recovery from an attack;
4. The selected studies generally propose solutions in a disjunctive way, that is, when proposing a routing solutions, for instance, they usually do so without regard to security or caching. The integrated solutions are scarce but desirable.

An equally not less important note is that, for each study the authors conclude that their own proposal generally presents better performance than the others for practically the same metrics. The availability of the source codes would be a good opportunity to better assess and globally compare the performance of existing solutions.

Design Overview

The present chapter aims at presenting the design, including the design decisions, of the proposed forwarding and routing model. The final model is composed by a forwarding strategy developed at the data plane level and the routing protocol, which is developed and works at the control plane.

This Chapter is organized as follows: The next Section 4.1 presents the context of the model design. Section 4.2 presents what have been done in order to develop VANET based on the NDN architecture. Section 4.3 presents the design of the forwarding strategy, which includes the presentation of the novelty of the model, in Section 4.3.1, the overview of the cached content and neighbor discovery process, in Section 4.3.2, the procedure that takes space when a Data packet is received, in Section 4.3.3, and when an Interest packet is received, in Section 4.3.4. Section 4.4 presents the model design of the routing model, which includes the presentation of its novelty in Section 4.4.1 and the initial route discovery, in Section 4.4.3. Section 4.4.4 presents the procedure for FIB management. The mobility prediction algorithms, STMP and LTMP, are presented in Section 4.5 and Section 4.6, respectively. Finally, a summary of the chapter is presented in Section 4.7.

4.1 Context

NDN is designed with routing and forwarding plans decoupled, which allow their development separately. Routing, in control plane, aims at determining optimal routes for content dissemination. Forwarding, in data plane, aims at using the provided routes, from the control plane, to forward the desirable content. In short, while the control plane directly manages FIB, the data plane does not. It only uses information from FIB, when there is any, to forward messages.

NDN can work without a routing protocol given the “intelligence” of the data plan, which can detect and recover by itself, from any situation of network failure, thanks to the stateful propriety of this plan in NDN. In fact, several studies including the ones presented in Section 3.11 take advantage of the fact that the data plane in NDN is stateful, and propose forwarding strategies that are based on flooding, instead of resorting to FIB to forward content. Uncontrolled flooding, however, may result in an undesirable problem,

the broadcast storm. As it is described later, several mechanisms are being proposed to avoid the broadcast storm. One of the known mechanisms is to select a specific node from an existing NT and unicast the packets to it. Only the selected node forwards the packets further. Nonetheless, the management of NT is another challenging issue. Several schemes resort to a beacon broadcast (a drawback) to discover new neighbors and for updating the list. Despite the aforementioned issues related to the maintenance of FIB, the need for routing is exhaustively investigated, and justified in studies such as [288, 261].

This chapter proposes a context and mobility-aware forwarding model for VANET based on NDN, and is based on an initial proposal in [230]. It updates the logic presented in the aforementioned manuscript and presents its evaluation. The aforementioned work proposes a combined model providing a forwarding and a routing protocol. Here, the model is presented separately, for data plane (Section 4.3) and control plane (Section 4.4).

4.2 Realization of NDN-Based VANET

Several solutions for realizing a NDN-based VANET have been proposed. Studies such as those presented in [277, 20, 77, 97, 29, 72], aimed at leveraging the characteristics of ICN/NDN for VANET. The study in [277] proposed a NDN-based inter-vehicular communication architecture aimed at improving content naming (supporting pull- and push-based communications), content addressing, Interest aggregation, and mobility in VANET. Arguing that the already proposed CCN extensions for VANETs still present issues related to scalability and mobility, [20] proposed a framework aiming at providing content-centric-based layer-3 and layer-4 operations over the IEEE 802.11p layers. The proposal leverages the CCN principles and takes further measures in order to better manage the node mobility and the broadcast-based wireless channel. The work in [77] highlighted the limitation of the NDN request-response communication pattern for the VANET event-based approach (e.g., the dissemination of traffic conditions for the better management of situations such as accidents) and, then, augmented the NDN architecture capabilities with Publish/Subscribe extensions. Similarly, [29] enhanced CCN with extensions in order to tackle the ad hoc propriety of VANET. Precisely, a special packet “event packet”, which is a push-based message aimed at disseminating emergency information, is added to the architecture, and FIB is modified in order to be able to select Faces depending on the requirements of the applications using them. The study in [97] proposed a prototype implementing a NDN-based VANET, which caches all received Data and uses geo-locations for content dissemination, with support for DTNs. Authors in [72] proposed a hierarchical, cellular, and cluster-based architecture for NDN-based VANET, where nodes dubbed “barycenters” take the responsibility for communicating with a cellular layer in each cluster. For the interested reader, several other studies can be found in surveys such as [46, 21, 208, 135, 13, 232], which include research challenges on implementing NDN-based VANET.

There are several proposed NDN-based VANET schemes for neighbor discovery and content dissemination, some of them are presented in Section 3.11. One of the main drawbacks of those proposals is the broadcasting of a beacon message for neighbor discovery and content dissemination [140]. Broadcasting a specific packet into the network increases traffic and can lead to network congestion. We propose CMAF, a protocol that, instead of broadcasting a beacon, leverages the overheard packets in the wireless channel to extract mobility information, later used to manage NT and to feed the mobility prediction algorithm. CMAF is designed to reduce the need for broadcasting a beacon, thus reducing the otherwise increased traffic in the network. The protocol is also designed to be context-aware, adapting itself according to the type of the received content (i.e., active safety, interactive/entertainment, comfort) and specific VANET environment (i.e., highway and urban), communication model (i.e., V2V and V2I), and network traffic (i.e., dense and sparse).

4.3 NDN Forwarding Plane: Context and Mobility-Aware Forwarding Model for V-NDN (CMAF)

The present section aims at presenting the overview of the proposed forwarding strategy, designed to work without resorting to the routing plane. It proposes a Context- and Mobility-Aware Forwarding (CMAF) model, which, instead of broadcasting a specific beacon for neighbor discovering, leverages the overheard packets to extract mobility information that is used to manage the list of neighbors and feed a mobility prediction algorithm. CMAF is designed to reduce the transmission overhead while keeping the performance high. The proposal is designed to avoid broadcast whenever possible, when running in V2V mode. In another configuration, running in V2I mode, we explore the in-network caching by means of an RSU-initiated process of sharing the list of cached content. The sharing procedure is only performed by nodes in the forwarding path from the RSU to the producer. In this configuration, we propose sending controlled periodic beacons from the RSU. The aforementioned procedure is explained later in this chapter.

4.3.1 Novelty of the Proposed Forwarding Model

State-of-the-art solutions resort to broadcasting a beacon for neighbors discovery. CMAF works in two distinct modes:

1. The beaconless mode (based on the V2V communication model) does not issue any specific Interest (beacon) for neighbor discovering. Instead, it leverages the proprieties of the wireless channel, where packets share the same channel and all can be sensed by the nodes in the communication range of the sender. To attain this, all packets carry NMSI of its sender. This way, the protocol avoids increasing the network load with additional packets for this specific purpose;

2. In the beacon-based mode (based on V2I communication model), beacons are periodically broadcast for content discovery. In this mode, an existing RSU sends a periodic Interest from which the producer responds with a special Data (sData) packet carrying a catalog of the contents of its CS by means of a BF. All intermediate nodes get to know which cached content the producer has and which cached content the previous nodes from the producer to the RSU have (this procedure is explained next, in Section 4.3.2). Additionally, a mobility prediction scheme proactively updates the neighbor's information in the NT. The mobility prediction scheme includes a proactive agent that is responsible for keeping the NT consistent (i.e., purging the inactive and out-of-transmission-range neighbors). The aforementioned agent periodically checks the status of the nodes in the NT. The neighbor purging of the NT takes place if any of the following conditions takes place:

- i) when its NMSI is updated and the computed distance to the current node falls out of the transmission range of this node;
- ii) when a time threshold ($t_{NTpurge}$) elapses without the current node hearing another packet from the neighbor in the NT, the corresponding neighbor's status changes to inactive. When the agent finds a node with the status "inactive", it means that the neighbor can no longer be selected as a relay; therefore, it is purged from the NT.

It is worth highlighting that, including additional information (i.e., NMSI) on the packets headers increases its size. An elongated packet is more likely to have transmission errors or collisions given that it takes more time to be transmitted. However, we anticipate that the advantages of including this additional information outweigh the beaconing alternative.

4.3.2 Overview of Cached Content and Neighbor Discovery Process

This section presents the content and neighbor discovery process. The specific forwarding process, for Data and Interest dissemination, is explained in Sections 4.3.3 and 4.3.4, respectively.

As aforementioned, the proposed protocol has two operational modes: a beaconless (V2V-based) and a beacon-based (V2I-based) model, for neighbor and cached content discovery.

4.3.2.1 Beaconless Mode

In this mode, no cached content discovery process takes place. The neighbor discovery leverages the broadcast-based wireless channel. The nodes build the NT based on packet overhearing on the air. The model is configured to operate in the promiscuous mode and process all received packets. When a packet (i.e., Interest or Data, including the unsolicited) is received, it follows the normal processing from NDN architecture and, additionally, the packet is sent to the NDN strategy layer, where the protocol processes it further. That is, all received packets are processed by the strategy layer.

First, the strategy layer extracts the NMSI from the received packet and builds/updates the NT. Two lists (NTs) are provided in each node, one for storing the list of mobile neighbors (i.e., NT) and another

- i.e., Neighborhood Table of RSUs (NTRsu), for storing the static neighbors (i.e., the RSU). Static nodes in NTRsu are persistently preserved in each node. The reasoning behind this choice is that given that the nodes are static, and knowing their geo-position, each time a node is in the vicinity of the geo-position of the RSU, it can always redirect the requests directly to that RSU, leveraging the broader knowledge of the network possessed by the RSUs. The NT management is provided by Algorithm 4.1. The NT management procedure is based on and resorts to (Line 19) the STMP algorithm, for predicting the next node's geo-location.

Algorithm 4.1: The NT insert or update procedure

```

Input : NMSI
1  n_geoPosition  $\leftarrow$  ExtractNeighborPosition (NMSI);
2  o_geoPosition  $\leftarrow$  GetOwnPosition ();
3  d  $\leftarrow$  CalculateDistanceToNeighbor (n_geoPosition, o_geoPosition);
4  neighbor  $\leftarrow$  setNeighborStatus (status_ACTIVE);
5  neighbor  $\leftarrow$  setNeighborDistanceToActualNode (d);

6  if (New neighbor is a RSU) then
7      insertNodeToNTRsu (neighbor) // A persistent insertion
8      insertNodeToNT (neighbor) // Possibly removed later when no packets are overheard from the RSU
9  else
10     if (Node exists in the NT) then
11         if (Timer still running) then
12             if (Timer has less than 100 ms to finish) then
13                 neighbor  $\leftarrow$  resetTimerForUpdatingNeighborInfo ();
14                 updateNT (neighbor);
15             else
16                 // Discards Update ()
17                 return;
18         else
19             if (neighborStatus is status_ACTIVE) then
20                 n_geoPosition  $\leftarrow$  PredictActualGeoPosition ();
21                 d  $\leftarrow$  CalculateDistanceToNeighbor (n_geoPosition, o_geoPosition);
22                 if (d less than transmission range) then
23                     neighbor  $\leftarrow$  updateNeighborDistanceToActualNode (d);
24                     neighbor  $\leftarrow$  resetTimerForUpdatingNeighborInfo ();
25                     updateNT (neighbor);
26                 else
27                     setNeighborStatus (status_INACTIVE);
28             else
29                 // Awaits For Removal From NT ()
30                 return;
31     else
32         neighbor  $\leftarrow$  setTimerForUpdatingNeighborInfo ();
33         insertNodeToNT (neighbor);

```

Every $t_{neighborUpdate} = 600$ ms, the STMP algorithm predicts the neighbors' location and updates the corresponding node's information in the NT. Whenever the mobility prediction takes place, the neighbor's status is updated. The status is set to *inactive* when the predicted geo-position is out of the current node's

transmission range and kept *active* when it is within the current node's transmission range. Moreover, every $t_{inactivePurge} = (3 \times t_{neighborUpdate} + 100)$ ms, the NT is checked, and all neighbors with status *inactive* are removed from NT.

The proposed STMP algorithm is locally linear. As explained in Section 4.5, we assume the time interval from the last packet received from a neighbor and the possible need to select this neighbor as relay, as considerably short. This assumption allows us to take the traveled distance also as short, and subsequently take the node mobility as partially linear. The $t_{neighborUpdate} = 600$ ms is chosen in order to predict the mobility every distance $d_{traveled} = 20 \text{ m/s} \times 0.6 \text{ s} = 12 \text{ m}$. That is, a prediction is made at most 24 m of the traveled distance with a maximum speed of 20 m/s, for two nodes in opposite directions. As it is around this distance (20 m) from a junction, that nodes begin to be incapable of receiving transmitted messages [169].

After the described process has occurred, the processing of the received packet continues as described in Section 4.3.3 for Data packets, and in Section 4.3.4 for Interest packets.

4.3.2.2 Beacon-Based Mode

In the urban scenario, the RSU sends, by flooding, periodic beacons for content discovery. The frequency of beacon sending is f_{RSUbf} . When a producer receives a beacon, it responds with the sData packet. The returned sData packet carries a BF that codifies the catalog of the contents stored in the node's CS. Each node receiving this packet extracts this information and forwards the packet further to the RSU. When a Data packet is returned, it carries its next-hop in the *destId* tag. So, besides the extraction of the BF, the node receiving the sData verifies if it has been registered/selected as the next-hop. Having been selected as next-hop, the node sends an unicast request to the previous node, from which the Data was received. The response to this unicast request is another Data packet (dubbed *sharingBF*) carrying a BF that codifies the list of this intermediate node's CS. All 1-hop away nodes receiving the *sharingBF* extract the BF and then drop the packet.

The mentioned procedure is explained in Figure 15. For instance, considering that figure, the intermediate nodes I_0, I_2, I_3 and I_4 all forward the beacon received from the RSU, and receive back the sData packet, which is processed to extract the BF it carries. The referred BF is a codified catalog of the content that the producer P has in its CS. When node I_3 receives the BF, it sends a request for a *sharingBF* Data packet to the node I_4 . The requesting Interest is received by all nodes (e.g., I_2 and I_4) in its transmission range, but only node I_4 responds to this request, as it is the final destination of it. When node I_4 responds with a *sharingBF*, all 1-hop nodes in its transmission range (e.g., I_3 and P) receive this packet and extract the catalog that codifies node I_4 's CS. This way, all 1-hop nodes from I_4 get to know the contents of its CS. The *sharingBF* Data packet is never requested to the producer, as it has already shared the sData packet, which includes the content catalog of its CS, by means of a BF.

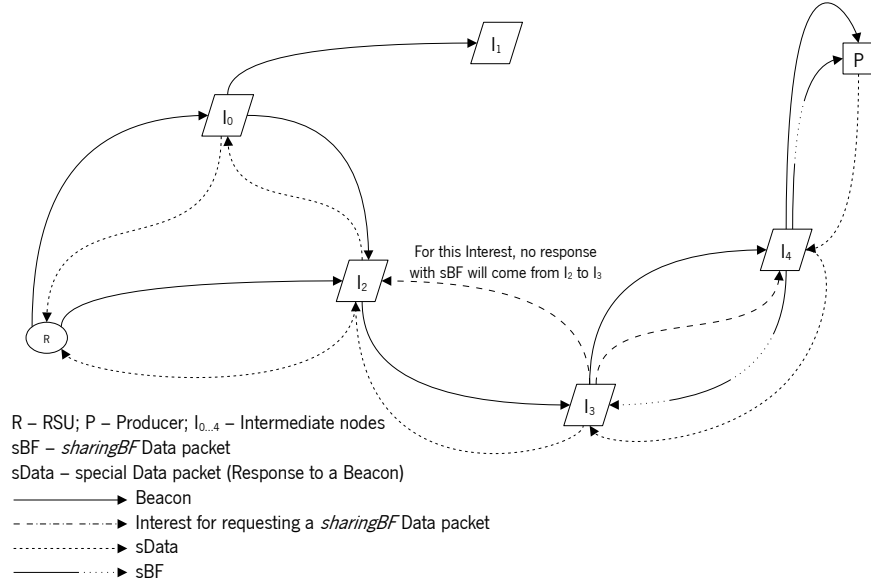


Figure 15: Beacon processing in intermediate nodes. All intermediate nodes receive and process the sData. Only 1-hop nodes from the sender receive and process the *sharingBF*.

In the urban scenario, after a first beacon has been received from the RSU, the existing vehicles are responsible for sending a beacon themselves with a frequency of $f_{vehiclebf} = f_{RSUbf}/5$. That is, after five times the periodicity of RSU beaconing any vehicle is able to send the beacon, and whenever a beacon is received, all vehicles reset their timers. Vehicle beaconing is only triggered if at least one beacon has been received from the RSU, meaning that an RSU is in the vicinity of the receiving node.

In fact, the described process has five objectives: 1) query the content providers and their content; 2) allow the intermediate nodes to know the content sources; 3) allow the intermediate nodes to know the RSU; 4) allow the nodes to update their list of neighbor nodes; and 5) use the *sharingBF* Data packet to distribute the list of cached content among intermediate nodes.

With this solution, a given node does not need to periodically broadcast beacon messages to query the list of cached contents from its neighbors. It only gather the list from 1-hop nodes with which it interacts for forwarding the sData. The cached contents list is stored as a Cached Content Table (CCT), which is, in fact, a list of BF, each one corresponding to a specific node that has previously interacted with the actual node for forwarding the sData. Each CCT entry is a tuple composed of the ID of the node from which a packet has been received, and the corresponding BF that codifies the catalog of the contents of its CS.

When a node receives a *sharingBF*, it extracts the received BF and the ID of the node from which the packet has been received. It looks up the received ID in the CCT. If the ID is not found in CCT, then no previous interaction with the node has occurred. Thus, a new entry is created, and the BF is stored. When the node ID is already registered in CCT, then a previous BF has been stored. If the previously stored BF is different from the received BF, then it is updated accordingly.

4.3.3 Processing of Incoming Data

Any intermediate node in the wireless network receives several packets, including the unsolicited Data. When a Data packet is received, CMAF processes it as presented in Algorithm 4.2. The first step is to check the application communication model, whether it is push- or pull-based (Lines 1 and 4). Then, the model looks in more detail to other application requirements (e.g., application type) as explained next.

Algorithm 4.2: Incoming Data processing

```

Input : Data
1 if (Data is Push-based) then
2   if (Data is still valid) then
3     Broadcast (Data);
4 else
5   if (Corresponding Interest exists in PIT) then
6     if (Data is Efficiency-related) then
7       Broadcast (Data);
8     else
9       InsertToCS (Data);
10      if (NT is not empty) then
11        threshold ← mean(NumberOfNeighbors_Last_3s());
12        hopCount ← getHopCount(Data);
13        if (hopCount is greater than threshold) then
14          Return;
15        nextHop ← SelectFromNTaNextHop_WithShortestDistanceToConsumer ();
16        UpdateNodePosition_STMP(nextHop);
17        SendData (Data, nextHop);
18      else
19        if (Data is Long-lived) then
20          InsertToCS (Data);

```

4.3.3.1 Awareness on Application Type

Active safety and efficiency related applications require the vehicle to be able to proactively share its information, without prior request. This is the case of push-based communication. With the stringent time constraints, and the highly dynamic topology, these applications cannot afford the routing procedures delays. These applications have a short-lived time, which would force the constant modification of the main structures of NDN (i.e., PIT, FIB and CS). Furthermore, as referred by [50], besides the fact that delays of warning messages should be low, in the milliseconds range, it should be ensured that all vehicles within a given area receive the messages in a timely manner. Therefore, we consider that a broadcast instead of routing is a better choice for this class. Existing proposals use different mechanisms to deal with push-based Data. Some of these mechanisms and their characteristics include those presented in the next studies.

Although the work in [249] states that the proposed communication model is V2V, the proposed name-based scheme clearly works only where an infrastructure exists, and is used to delimit the region where broadcast should take place. This proposal relies on naming for selecting the push-based among

all the unsolicited Data. Resorting to naming, the way the design is performed, as well as the subsequent prefix match, as proposed in this study, restricts somehow the dissemination of push-based Data in the V2V scenario. The proposal by [166] relies on beacon messages to initiate the processing of push-based Data. The beacon messages are used to guide consumers to prepare a PIT entry that should be used to accept the Data to be sent to the network. This scheme floods the network with a beacon, prior to sending push-based content. Resource usage is not optimized with this procedure. The authors in [25] propose three strategies for push-based Data in Internet of Things (IoT): (1) Interest notification, appending small Data to Interest, modifies the semantics of the Interest packet. (2) With unsolicited, all Data packets are accepted and stored in CS as proposed in [97]. Then, like in the previous strategy, the consumer broadcasts an acknowledgment packet, confirming the reception of the Data. (3) Virtual Interest polling, based on long-lived Interest, locks a PIT entry for a long time. Using additional broadcast messages to prepare the node on the reception of an unsolicited Data as proposed in [166, 25] leads to increased network traffic. The study by [241] proposes an Android-based application for named-Data emergency network services, with a push-based Data strategy. The mentioned strategy allows the forwarding plane to process a Data packet which does not have a corresponding PIT entry.

Push-Based Data Although our proposal could use a scheme similar to [241], which clearly distinguishes the push-based content (which we consider for active safety and some efficiency-related applications) and the other unsolicited Data packet that can arrive at a given node, we propose a generic name-based scheme to classify the contents, and treat differently contents falling into the active safety class. CMAF proposes a scheme where all active safety-related contents are based on the following prefix:

/push-based/info-type/sender-ID/sender-geo-coordinates/

The first component is used to identify the type of content as being push-based and destined for broadcast. We can have different types of push-based content. For instance, besides the active safety content, we have the content destined for road efficiency. The latter is longer lived than the former, thus, we can treat it differently from the former. This is the reasoning behind the distinction that will be provided by the second component.

The third component is the identification of the sender, and the last component is the geographical location of the sender. We consider it important to have the geographical coordinates of the sender, considering that active safety content is location dependent. The broadcast information is only important a hundred meters away from the location where it has been sent.

If the received Data packet is push based, the model checks whether the Data is still valid. If still valid, it is broadcast. As referred before, broadcasting is performed in order to shorten the delay that would exist if a routing procedure were to take place. If Data validity has elapsed, then the packet is discarded.

Only applications in the active safety and efficiency classes can push Data into the network. Applications in the comfort and interactive-entertainment classes can provide Data only by request. Some

applications in the efficiency class can also provide Data by request. This is the case of pull-based communication, for which the forwarding procedure is described next.

Pull-Based Data Similar to the original procedure in NDN, when Data is received, a corresponding PIT entry is looked up. If a corresponding PIT entry does not exist, then the content is unsolicited. To take advantage of the less stringent characteristic of VANET in terms of CS size, unsolicited content is cached, similar to [97]. If a PIT entry exists, then the message was requested. The next step is to check if the Data falls into the efficiency class, in which case the Data is broadcast, as explained before. As the content of some efficiency-related applications may live longer, then, it is cached. The long-lived Data application includes, for instance, the audio/video downloaded. The behavior of caching all long-lived content also favors the DTN.

If the received Data is classified into the comfort or interactive-entertainment classes, then the original NDN procedure takes place, i.e., its corresponding FIB entry is updated, and Data is added to the CS and then forwarded downstream. The Data packet is forwarded downstream using breadcrumbs from PIT entry. As observed by [95], the RTT 95th percentile for an Interest–Data transaction is less than 300 ms. In the referred period, the vehicles do not move far between the time an Interest is issued and the time the corresponding Data packet is received, thus ensuring the validity of the Interest breadcrumbs in the PIT and the effective retrieval of Data packets. Therefore, CMAF assumes that even considering the topology is dynamic, the created paths from upstream forwarding will have enough stability for Data forwarding, mainly for urban scenarios where the speed of vehicles is relatively low.

Now, even considering the aforementioned assumption about the relative stability of routes, the link duration in VANET is relatively short, and the topology is continuously changing, which result in a frequent network disruption [213]. In the case of route disruption, the last node sending the packet gets, from the Interest, the geographical coordinates of the consumer and can predict the new consumer location, and a possible route to use for forwarding the Data to it. The updated NT is checked to determine the better relay node for Data delivery. Differently from other solutions, in which the Data packet is always forwarded whether there is or not a neighbor for forwarding it further, in CMAF, the node only sends the Data packet when there is at least one *active* neighbor. This way, the network resources are only used when forwarding to the next-hop is guaranteed.

Figure 16 depicts the overview, at the higher level, of the incoming Data processing. The “Forward Data” block refers the process performed from Line 10 of the Algorithm 4.2. The routing procedure, in the “Update FIB” block, is performed later with MARV. At this level, FIB is not explicitly managed. Also, it includes a block related to the CCT management, which only works in urban environments.

4.3.3.2 Awareness on Network Density

Studies such as [97] limits the hop count for a fixed value. The number of hops a packet traverses from its origin to the destination depends on the network density. Therefore, limiting the packet forwarding for

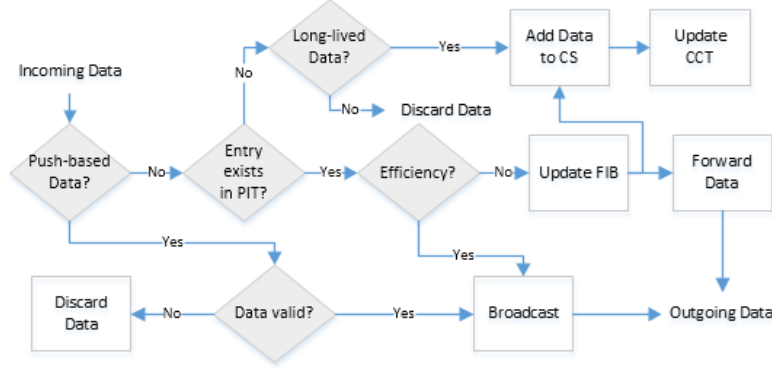


Figure 16: High level overview of Data processing in the intermediate nodes

a fixed value for different network densities may not be a better solution. CMAF is designed to be aware of the network density. Differently from the aforementioned study, we propose to limit the traversed hops based on the current number of neighbors a given intermediate node has. Before the packet is sent out, downstream, the model computes the average number of neighbors (used as a *threshold*) the node has encountered for a previous period of $t_{hopThreshold} = 3000$ ms (Algorithm 4.2, Line 11). This value is chosen from simulations. When the received packet has already traversed a number of hops superior to the *threshold*, the packet is discarded, and the consumer eventually sends the request again, which can hopefully be satisfied by any node in its vicinity that has received the packet before. In fact, from simulations, the *threshold* is corrected to a new ($threshold^+$), by the Equation (4.1) when $threshold > \alpha$:

$$threshold^+ = threshold - \beta \quad (4.1)$$

The chosen values are $\alpha = 14$ and $\beta = 2$. We highlight that only Data packets have their hops limited. This is due the fact that flooding Data packets is more disastrous than flooding Interests, given their bigger size. Moreover, the model allows Interest flooding in order to leverage the caching system, that is, even if the requested Data do not reach the requester due to the hop limit, the next time the same Data are requested, they can be fetched from a nearby node that can have them already cached from the previous request.

4.3.4 Procedure for Incoming Interest

Algorithm 4.3 presents the processing of an incoming Interest. When a Interest is received, it is inserted into PIT. As the sender node is 1-hop away from the receiver, the node which is taken as a neighbor, gets its identification (including its NMSI) inserted (if new) or its NMSI updated in the corresponding NT entry as explained by Algorithm 4.1 in Section 4.3.2.1.

After updating the NT, it is verified whether the received Interest is duplicated or not. If it is duplicated, a procedure for looped Interest takes place. Otherwise, it is verified if it is a pending Interest. If pending,

Algorithm 4.3: Incoming Interest processing

```

Input : Interest
1  InsertOrUpdatePIT (Interest);
2  InsertOrUpdateNT (Interest);
3  if (Interest from 1-hop RSU) then
4    InsertOrUpdateNTRsu (Interest);
5  if (Corresponding Data in CS) then
6    ReturnToRequester (Data, destId);
7  else
8    if (Destined to RSU) then
9      SendInterestToRSU (Interest);
10   else
11     if (Efficiency or push-based) then
12       BroadcastInterest (Interest);
13     else
14       if (nextHop is known) then
15         if ((NT not empty) or (NTRsu not empty)) then
16           neighbors ← getListOfNeighbors();
17           UpdateNodePosition_STMP(nextHop);
18           for (each nextHop in nextHops) do
19             if (nextHop in neighbors) then
20               SendInterest (Interest, nextHop);
21             else
22               neighbor ← NeighborWithMinDistanceToNextHop (neighbors);
23               UpdateNodePosition_STMP(neighbor);
24               SendInterest (Interest, neighbor);
25         else
26           // FIB is empty or does not have a corresponding Interest entry
27           if (Interest in CCT) then
28             nextHop ← NextHopFromCCT ();
29             if (NT is not empty) then
30               neighbors ← getListOfNeighbors();
31               if (nextHop in neighbors) then
32                 UpdateNodePosition_STMP(nextHop);
33                 SendInterest (Interest, nextHop);
34               else
35                 neighbor ← NeighborWithMinDistanceToNextHop (neighbors);
36                 UpdateNodePosition_STMP(neighbor);
37                 SendInterest (Interest, neighbor);
38             else
39               if (NTRsu is not empty and NTRsu in range) then
40                 UpdateNodePosition_STMP(RSU);
41                 SendInterest (Interest, RSU);
42             else if (NT is not empty) then
43               BroadcastInterest (Interest);

```

then a CS miss has occurred. If not pending, then a CS lookup takes place. In a case of CS hit, the corresponding Data packet is returned. In the case of CS miss, it is verified whether the Interest is explicitly destined to a RSU or not. If it is indeed destined to a RSU, then it is sent to the selected RSU.

If the Interest is not marked as destined for a specific RSU, then it is checked if it comes from an efficiency-related application, Line 11. If it comes from an efficiency-related application, then it is

immediately broadcast. The reason for the broadcast decision is explained in Section 4.3.3.1.

If the Interest cannot be satisfied by the present node, then its corresponding Data source still needs to be discovered. Existing solutions follow an almost similar procedure from this stage on, which is the immediate Interest broadcasting for content discovery. The study by [125], for instance, which uses FIB, follows a slightly different mechanism. The Interest is forwarded towards nearby RSU, which is supposed to have knowledge about other routes. If routes are not found even on RSU, the requester node delays its requests and awaits the creation of FIB entry, from a general beaconing process. CMAF deals with contents related to different applications classes differently. But, for comfort and interactive-entertainment applications, it follows a similar mechanism to that of the mentioned study. However, differently from the referred study, the proposed model does not await the creation of an FIB entry (CMAF does not use FIB). Instead, it broadcasts the Interest if there is any neighbor around. Before broadcasting the Interest, however, a last feature is explored, which is the use of known cached contents (i.e., see CCT, in Section 5.4) from other nodes.

Only if the content is not found in CCT, the broadcast procedure for content discovery takes place. If the corresponding Data are found in this table, then an updated route (i.e., geo-location) to the content is used, and the Interest is forwarded to the node holding the content. The broadcast for content discovery is performed in two different ways, depending on the current network scenario, as described in Section 4.3.4.2.

Similarly we present in Figure 17 the overview, at the higher level, of the incoming Interest processing. The “Forward Interest” block includes the processing of NT and NTRsu, as presented by the Algorithm 4.3. At this level, FIB is not explicitly managed, thus the “Add Interest FIB entry” block is not used yet. It also includes a block related to the CCT management, which only works in urban environments. The figure is included just to show what the final model looks like, and help clarify what this section presents.

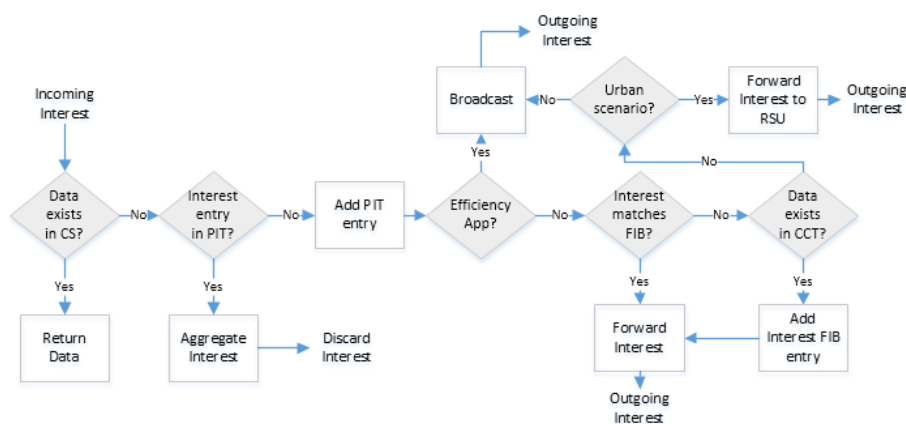


Figure 17: High level overview of Interest processing in the intermediate nodes

4.3.4.1 Forwarding Enhanced By Caching

To the best of our knowledge, the existing proposals for NDN-based forwarding for VANET do not usually explore caching for deciding where to forward packets. The sole exception found is the study by [80]. The aforementioned study explores caching by inquiring into the contents in the neighbor's cache. Vehicles know the neighbors' cached contents by periodically notifying them of their local cache information. This notification is performed via broadcasting beacon messages. All nodes broadcast the beacon.

Considering that periodic beacon broadcasting, by all nodes, increases network traffic and subsequently increases congestion and potential collision, CMAF proposes a different solution. In this solution, a producer keeps an updated list of its local cached content, and will append it to the sData packet via a BF, which is sent by intermediate nodes in the forwarding path from the content source to the RSU, in response to a beacon sent by the RSU (process explained in Section 4.3.2.2). The aforementioned process is only considered for urban scenarios where the network includes a RSU. Another use of cached content in the intermediate node is also explained in Section 4.3.2.2 and is related to the use of CCT and the *sharingBF* Data packet.

As referred before, the model proposed in this work leverages the less stringent characteristics of VANET, concerning the storage size, and caches all received Data. But, differently from other works (e.g., [97]), the present model caches unsolicited Data based on their classification in terms of application type. The short-lived content from the active safety application class is not cached. Some long-lived content from the efficiency application class are cached along with the long-lived content from the interactive-entertainment- and comfort-related applications.

The reasoning behind the decision to avoid the caching of some short-lived content can be reinforced by the works in [24, 22], which study and propose caching strategies based on the content lifetime. As stated in the referenced works, the CS space is not efficiently used when caching contents that are then shortly removed.

4.3.4.2 Awareness on Network Scenario

VANET is highly dynamic with frequent partitioning, having relatively short inter-contact times. To deal with this partitioning, it is necessary to build protocols that can efficiently adapt between highly disconnected, dynamically partitioning networks and non-partitioned networks [213]. Vehicles traveling between these scenarios need to adapt their behavior to the current network characteristics in order to provide high QoS and QoE to the users.

Having in mind the description presented in Section 4.3.2 for content discovery, when a consumer or any intermediate node does not find matching Data in its CCT, the node proceeds in two different modes according to the current scenario as detailed next.

For urban scenario where vehicles move at relatively lower speeds and there exists a higher density of vehicles and infrastructures, broadcasting has a more negative impact than in the rural scenario. Thus, as all nodes have a list of all existing RSUs in the network, and the RSUs have better knowledge of routes

to the content sources (given their periodic beaconing to discover contents as described in Section 4.3.2), the proposed model prioritizes the V2I network communication model and redirects the Interest to the infrastructure (i.e., RSU). It is a matter of preference, not a limitation in only one communication model. If before reaching the RSU, a route to the content source is found in any of the intermediate nodes, then from this node, the request is unicasted using the new route. Otherwise, the Interest forwarding continues until it reaches the RSU, and from there, it is hopefully forwarded to a known content source. If no content source is known from the RSU (i.e., not even a CCT entry exists), then it is broadcast. The Interest is only broadcast as the last resort. The described procedure is performed in order to reduce the Interest broadcasting.

In a rural scenario, the preferred network communication model is V2V. Having in mind that this environment is less dense in terms of infrastructure and vehicles, thus, a broadcast is less problematic. Therefore, being the last solution left, a broadcast is used.

As presented in Section 3.11, state-of-the-art solutions adopt a timer- and contention-based mechanism for tackling the broadcast storm problem induced by allowing all nodes to forward the same received Interest. The time-based mechanism as presented by those solutions allows the nodes to send an Interest, one node at a time, and the other nodes suppress the forwarding when overhearing the same packet from another node. Some of these solutions disseminate the Interest in one direction and others in two directions, one for each road direction. CMAF proposes a different approach. In order to ensure the dissemination of the received Interest in all directions, all nodes receiving an Interest immediately forwards it and then stores it in a timer-based container with a duration of order of magnitude of the retransmission time (i.e., $t_{RTX} = 3$ s). The first Interest is always forwarded. If the same Interest (i.e., same sequence number) is received before the timer elapses, it is dropped, and the container timer is reset.

4.4 NDN Routing Plane: Mobility-Aware Routing Model for V-NDN (MARV)

As mentioned before, the main objective of routing, from the control plane, is to determine optimal routes for content dissemination. That is, it populates and manages FIB and in coordination with the forwarding strategy, from the data plane, it makes the assessment and testing of the forwarding interfaces.

The constant route disruption, due to high mobility of nodes in VANET, render the task of maintaining routes in routing tables challenging. The previous statement is further reinforced by works such as [251, 97, 239, 21] which argue that maintaining an updated FIB is computationally costly and not feasible at all, and that a flooding-based approach is the one that better suits VANET. As explained in Section 2.2.2, although NDN can work without a routing protocol given the intelligence of the forwarding plane, which can detect and recover by itself from any situation of network failure, the need for routing is exhaustively investigated and justified in works such as [288]. The justification provided by [288] is partially further

backed up by the work in [261] which investigated the synergies between routing and forwarding strategies in NDN. The aforementioned studies are based on wired NDN. Nevertheless, even considering the aforementioned assertions, several routing proposals for wireless networks, such as VANET, have been proposed - see the state-of-the-art in Chapter 3. The results presented by such studies show that for specific scenarios routing can still be feasible and useful in VANET. The main objectives of these solutions include: a) the elimination of the broadcast storm problem posed by the flooding approaches; b) the optimization of resource usage, by means of reducing TO, which results in c) the improvement of data retrievability. The model proposed by this thesis, dubbed MARV, is also intended to contribute on achieving these objectives.

MARV is a routing protocol that resorts to mobility prediction, and aiming mainly at minimizing TO as an essential step for maximizing throughput, and thus, increase the network efficiency. It is also designed to conform with the functional requirements for a good routing protocol [200]: a) present a low latency; b) present the capability of adapting its addressing mode to unicast, broadcast and multicast; c) present high delivery rates; d) at most, cause low additional load; and e) adapt itself to the specific VANET environments (i.e., highway, suburban, urban).

4.4.1 Novelty of the Proposed Routing Model

When dealing with routing in a highly mobile network such is the case of VANET, the main issue to deal with is the frequent partitioning and route disruption which require constant FIB updates, at a risk of ending up using dead routes. Additionally, the size of FIB should be kept within a manageable size, otherwise the prefix's lookup process may become burdensome and end up requiring the use of complex and resource's demanding algorithms.

MARV deals with the mentioned issues as follows:

- In order to keep FIB updated, state-of-the-art solutions (e.g., [214, 80, 76]) resort to either broadcasting a beacon to continuously query the content sources or allow the producer to periodically announce its prefixes (e.g., [293, 292]). MARV proactively and periodically updates FIB by resorting to mobility prediction.
- To keep FIB size manageable, some solutions periodically delete the entire FIB and regenerate it whenever a newly requested content is received from neighboring nodes (e.g., [129]). Instead of deleting the entire FIB, MARV uses a procedure described in Section 4.4.4 for avoiding frequent beaconing to update FIB and to assure a manageable size of it, without the need to regenerate it frequently.

4.4.2 MARV Architecture Description

Depending on the chosen grouping criteria, routing protocols in VANET can be categorized in topology-based and position-based (geographic) or by the addressing mode in unicast, multicast (includes geocast

and cluster-based), and broadcast [200, 223, 49].

Topology-based protocols require the knowledge of the network topology before they can make decisions on routing. This category can be further divided into reactive, proactive, and hybrid protocol subcategories. Proactive protocols build and maintain an up to date routing table with existing routes to the content sources. A timely beacon broadcast is an usual adopted mechanism to maintain updated the routes, and discover new paths to the content sources. In a highly mobile environment, routes demand frequent updating. This process can result in an increased network traffic overhead. Reactive protocols in the other hand, need only to maintain a limited number of routes in the routing tables, because the routing and discovery process only takes place when the node requires a path to the content source. That is, the route is established when its is needed. The hybrid subcategory takes synergies from the two aforementioned approaches. Due to the highly changing topology in VANET, topology-based protocols are not well suited for this network [177, 257].

In contrast, position-based (or geographic-based) routing protocols do not need to know the network topology or perform a prior route discovery. Instead, they use the position/location to forward packets hop-by-hop towards the destination. The position/location of the nodes is gathered by means of navigational systems on-board, such as the GPS [44]. Given that position-based routing protocols do not need to know the network topology, i.e., do not need to store entire routes to the content providers (may only require to keep a list of 1-hop neighbor), they are more scalable and more suited for highly mobile environments. Position-based routing protocols are reactive, in the sense that the next-hop node is selected on demand, taking in account aspects such as, the geographical distance to the content provider. This category may be grouped into the DTN, Non-DTN and hybrid.

The RQ1 in Chapter 3 resulted in an extensive presentation of the state-of-the-art for routing in VANET. Furthermore, the aforementioned chapter proposed what was dubbed as a “Possible roadmap for further developments”. In these terms, the solution proposed in this section, MARV, is based on the proposed roadmap. In summary, MARV resorts to mobility prediction for tracking and selecting best routes to the content providers. The presented solution is mobility-aware in the sense that its design is based on the node mobility knowledge. Moreover, MARV is designed to explore multi-path toward the content provider.

Taking in consideration the combination of taxonomy and classification proposed in [122, 78, 293] and Table 13, which presents the extracted data for state-of-the-art, MARV is an hybrid model designed to work both in reactive and proactive modes. As aforementioned proactive routing protocols present the disadvantage of processing overhead due to computational processing demand for updating nodes mobility, and reactive routing protocols present the disadvantage of high delays due to the fact that routes to content providers are processed on demand when the content is requested. Being hybrid, MARV is designed to provide a balance for the aforementioned issues (e.g., high computational processing and high delays).

The routing algorithm - Algorithm 4.4, is a non-ending, and timer-based, procedure which updates the existing FIB entries resorting to the mobility prediction algorithms.

Algorithm 4.4: Routing process

```

Input : Data packet; Data-Prefix;
1 timer ← setTimer(1900ms);
2 if (incomingData == True) then
3   if (InterestSatisfied == True) then
4     Nmsi ← getNmsiFromData();
5     FIB.insertOrAggregateNextHop(InterestPrefix, NMSI); //For multipath purpose, only "threshold" next-hop with the
        lowest cost are kept;

6 //Time triggered mobility prediction and FIB update;
7 if (timer elapsed) then
8   foreach (next-hop in FIB) do
9     [predictedPosX, predictedPosY] ← stmp(next-hop);
10    if (next-hop in NT) then
11      // next-hop is still a neighbor;
12      FIB.UpdateNextHop(predictedPosX, predictedPosY);
13    else
14      FIB.RemoveNextHop(next-hop);

```

4.4.3 Initial Route Discovery

The first Interest received has no corresponding FIB entry, thus, similarly to the solutions in works such as [128, 26], CMAF broadcasts it for content discovery. Before broadcasting the aforesaid Interest, the node assures that it does not have any RSU registered in its NTRsu or, having any RSU already registered, there is none in its vicinity. The model communicates in V2V. If there is any RSU, the model commutes to V2I mode and instead of broadcasting, the node unicasts the Interest to the existing RSU(s). This procedure leverages the broader knowledge of the RSU(s) about the network. When a Data packet returns, all nodes in the forwarding path update their FIB and new unicast paths to the content provider are, then, created.

As aforementioned all NDN packets carry an additional information - NMSI, used by all nodes to extract information related to the mobility of the nodes from which the packet was received. The information carried in NMSI is stored in FIB.

The NDN architecture was originally designed for wired networks, where neighbors of a given node are unambiguously identified by the Face they are connected (wired) to. Sending or receiving a given packet to or from a specific node is achieved by selecting the Face the node is connected to. PIT is extensively used to support this model, where an entry is created for each upstream and non-duplicated incoming Interest, associated with its ingress Face. However, different from the wired channel which is unicast, the wireless channel is broadcast-based by nature, i.e., each node within the communication range of the sender overhears the packets in the channel. Thus, for wireless channels, the Face-based communication model is not feasible. A manner to unambiguously identify a node in wireless-based networks, for unicast and anycast communications, is crucial [50]. A good candidate for node identification is the MAC address of the net devices, installed on these nodes. Some representative studies that use MAC addresses to identify nodes include [192, 133, 129, 125].

When an Interest is satisfied, a corresponding entry is created in FIB (see block from Line 2). For

a wired network, as refereed in the last paragraph, the Face from which the Data was received is also registered as a next-hop for future requests. For wireless scenarios, the Face used by all communications between that specific node and its neighbors will, typically, be the same. In that case, as it is in the solution proposed by this thesis, the MAC of the sender is registered alongside the Face ID. The referred MAC is included in the NMSI carried by the received packet. The *cost*, a already existing field, is updated with the number of the hops traversed by the received packet. If, meanwhile, another node satisfies the requested content, a new next-hop is registered. As the received content corresponds to the same name-prefix, then instead of creating a new FIB entry, the previous is updated with the new next-hop. That is, the new next-hop is aggregated (see Line 5) to the previous corresponding FIB entry.

4.4.4 FIB's Management Procedure

In dense networks, FIB can grow immensely and aggravate the complexity of the lookup process. A means for periodically purge dead routes and reduce FIB size is mandatory. Given that the proposed model keeps an updated list of neighbors, as described in Section 4.3.2.1, the next-hops in FIB are periodically compared to the active neighbors in the list (see block from Line 10). If a next-hop exists in FIB but it is not listed as active in the NT, then it is removed from FIB. This procedure takes place every $t_{NTpurge} = 1900$ ms. This value is chosen as explained in Section 4.3.2. Before the purging procedure takes place MARV uses STMP (Section 4.5) in mobile nodes and LTMP (Section 4.6) in static nodes, to update the geo-location of each next-hop registered in FIB.

For the sake of multi-path routing, the number of routes (next-hops), for each name-prefix, is limited by the threshold defined by Equation 4.1, which is based on the network density, as seen by the number of active neighbors (performed by the method in Line 5). Only the next-hops with the lowest cost are kept in FIB. The ones with the lowest cost are chosen considering that they are the nearest ones, so the link stability to them should be the highest. Moreover, next-hops with high costs could mean the route most probably has a loop.

From here, the process is handed to the data plane which proceeds as described in Section 4.3.4.

4.5 Short-Term Mobility Prediction (STMP)

Aiming at reducing flooding for content and neighbor discovery (or to find the new position of the moved content source) and consequently reduce network congestion, delivery delay, and packet loss, CMAF resorts to mobility prediction for tracking the geographical position of the moving nodes. For this regard, new specific tables (see Section 5.4) are included for managing neighbor nodes, including their geographical positions, and the existing content in the neighbor's CS. These tables are periodically updated by the mobility prediction algorithms. Moreover, leveraging the in-network caching, the proposed solution takes advantage of vehicles moving between disconnected regions to opportunistically distribute contents between these regions as "content mules".

The key characteristics of VANET (i.e., highly dynamic topology, and intermittency of connectivity) make it difficult to build and maintain FIB [97, 239]. Although these characteristics impose specific challenges for several applications, they can be exploited, for instance, for developing better movement management. In fact, due to the movement constraints imposed by the road topology and traffic conditions combined with the repetitive characteristic of users' movements, the future movements of the vehicles are usually predictable. As concluded by [155] on the predictability limits of vehicular mobility, there exists some strong regularity in the daily vehicular mobility in both the temporal and spatial dimensions. This regularity can be exploited to predict the vehicular mobility with a high degree of prediction accuracy.

Some representative applications that take advantage of prediction algorithms in VANET include routing, forwarding, traffic management, road safety, handover management, resource management, and location-based applications [2, 298]. Based on their objectives, the prediction algorithms can be categorized into (1) link stability prediction; 2) location prediction; 3) trajectory prediction; 4) traveling time prediction; and 5) collision prediction. See [2] for details.

The routing and forwarding model we propose uses the STMP algorithm (i.e., only a future location prediction) for updating NT in all vehicles. The STMP algorithm is based on the EKF formulation presented in [207]. The use of STMP makes sense mainly in an urban scenario, where the density of the infrastructures and road topology constrains the duration of inter-vehicle link, which forces the need for frequently updating the nodes' NTs.

In CMAF, we adopt EKF assuming that the time interval from the last packet received from a neighbor, and the possible need to select this neighbor as a relay, will be considerably short. Our reasoning is that if the interval is short, we can also consider the traveled distance to be short, and subsequently assume the mobility to be partially linear or easily transformed to non-linear. This way, we keep the processing cost in vehicles low.

The proposed STMP is presented in Algorithm 4.5.

4.6 Long-term mobility prediction (LTMP)

Although for the sake of autonomy, VANET should be able to work independently from the infrastructure, we foresee a crucial importance of RSU whenever they exist, for a yet better performance, and better management of the network resources. We assume that RSUs can have a better view of the network topology under their transmission range. Hence, when the vehicles have no FIB entry toward the content provider, instead of broadcasting the Interest, our model prioritize forwarding the packet to the nearest RSU, which have a better prediction of the content provider current location. The RSU prediction is based on LTMP or on the information provided from the periodic beaconing. As explained in Section 4.5, the vehicles will explore a more light mobility prediction approach, in order to save computational resources. Assuming that RSU will have more dedicated resources, a more elaborated and complex mobility prediction will be running at the RSUs. Considering the results presented in Section 2.4, we adopt the HMM (presented in

Algorithm 4.5: The STMP algorithm, based on EKF

```

Input : Neighbor
Output: UpdatedNeighbor, distance

1  neighborId ← getNodeId(Neighbor);
2  prev_timestamp ← getTimestamp(nmsi);
3  neighborTemp ← Neighbor();
4  values, sizeValue ← getMobilityHistoryValues(); // We used N = 4, historical values;
5  if sizeValue is 0 then
6      timestamp ← getCurrentTimestamp();
7      Node_position ← NodeLocation(current_Node);
8      posX, posY ← NodeLocation(Neighbor);
9      speedX, speedY ← NodeSpeed(Neighbor);
10     distance ← CalculateDistance(Node_position, posX, posY);
11     neighborTemp ← Neighbor(posX, posY, speedX, speedY, distance, timestamp);
12     return neighborTemp, distance;

13 if sizeValue > 1 then
14     for i ← 0 to sizeValue - 2 do
15         posX, posY, speedX, speedY, timestamp2 ← values[i];
16         tmpVector ← [posX, posY, speedX, speedY];
17         v ← RadarCartesianToPolar2(tmpVector);
18         ListMeasurement.push_back(Measurements{v[0], v[1], v[2], timestamp2});
19         ListGround.push_back(GroundValues{posX, posY, speedX, speedY});

20 else
21     // Processing the single value case;
22     posX, posY, speedX, speedY, timestamp2 ← values.at(0);
23     tmpVector ← [posX, posY, speedX, speedY];
24     v ← RadarCartesianToPolar2(tmpVector);
25     ListMeasurement.push_back(Measurements{v[0], v[1], v[2], timestamp2});
26     ListGround.push_back(GroundValues{posX, posY, speedX, speedY});

27 PackListMeasurement.push_back(neighborTemp);
28 Initialize_trackingKFLasteEstimate();
29 for k ← 0 to N-1 do
30     m_trackingKF.ProcessMeasurement(PackListMeasurement[k], PackListGround[k]);
31     if k is N-1 then
32         if trackingKFLasteEstimate ≠ m_trackingKF then
33             position ← getCurrentNodeLocation(node);
34             distance_n ← CalculateDistance(position.x, position.y, m_trackingKF.ekf_x_(0), m_trackingKF.ekf_x_(1));
35             neighborTemp ← updatedNeighborWithKFEstimates();
36             Add to estimations and ground_truth;
37             trackingKFLasteEstimate ← m_trackingKF;
38         else
39             Update_trackingKFLasteEstimate();

40 return tuple(neighborTemp, distance_n);

```

2.4.3) to design our model.

Authors in [271] noted that, current mobility prediction methods typically consider the spatial and temporal spaces independently. However, correlating spatial and temporal information is a key aspect in mobility prediction methods based on user's history. For instance, the proposal by [279], as presented in Section 2.4, considers only the spatial dimension. The authors propose a V2X routing for VANET, with mobility prediction based on HMM. In that proposal, the road is divided into discrete segments. The road segments terminate at intersections or dead ends. A *link point* is defined, which represents the location and current state of a vehicle in terms of road, speed, and direction. A *trip sequence* is also defined. It

represents a collection of movement records corresponding to the observation sequence. The proposal considers a differentiation in two types of position of a packed, a destination and an intermediate road segment. Having in mind this division, and considering that the final position of a vehicle in movement is unknown, the final position is considered as a hidden state and the immediate road segment to reach by a vehicle is the observable state.

We propose a mobility prediction model somehow similar to this study but, we extend it to include the temporal dimension. We propose a temporal-spatial dimension, thus including for a more accurate prediction, the time in which a vehicle traversed a particular segment or each intermediate intersection/destination. To include temporal dimension we can adopt the proposal used in [173] on predicting human mobility. That study proposes a method based on HMM to predict human mobility. In the proposed solution, the human location histories are first clustered according to their temporal period, and in which the clusters are then used to train different HMM, one for each cluster. Specifically, they create three fixed clusters: 1) Weekdays, from 7AM to 7PM; 2) Weekdays, from 7PM to 7AM; and 3) Weekends. They use the location characteristics as unobservable parameters, and also with the effects of each individual's previous actions. The experiment results, using a dataset from GeoLife project, is not satisfactory, with prediction accuracy of 13.85%. Having in mind that pedestrian networks data are more detailed and complex than vehicular datasets [34], and assuming that vehicular mobility is more predictable than the human mobility, due to it being more constrained (by the mobility models, and mainly by the road topology), even though the results achieved by that study are not satisfactory for predicting human mobility, an improvement of their idea, applied to VANET could result in better performance. Our model could, instead of fixing the granularity to three clusters, create dynamic clusters depending on each user mobility characteristics (in temporal terms). This is a proposal that could be explored.

The need to correlate the spatial and temporal dimensions can be explained by the following example, Figure 18.



Figure 18: Hypothetical user's mobility trace

Based on this figure, suppose a user has the following hypothetical weekly schedule, Table 19. The presented schedule is simplistic and used just to clarify our idea. If the time dimension is not considered,

then for Activity A_3 and A_9 , for instance, which follow the same three initial intermediate roads (intersections), $s_6 \rightarrow s_8 \rightarrow s_7$ ($x_4 \rightarrow x_3 \rightarrow x_2 \rightarrow x_1$), we have to resort to only the number of times the user took the road s_2 ($x_1 \rightarrow x_6$) or s_1 ($x_1 \rightarrow x_0$), whereas, using the time we can calculate the probability considering the time on the day, that the user is heading to his/her destination. For this present case, if a user has followed the roads (intersections) $s_6 \rightarrow s_8 \rightarrow s_7$ ($x_4 \rightarrow x_3 \rightarrow x_2 \rightarrow x_1$), before the midday then the probability of using s_2 , toward x_6 is high. If it is in the afternoon or later, then the user will, with high probability, follow the road s_1 , toward the intersection (destination) x_0 . This is much important if we recall that the future state of Markovian models is only related to the current state, not considering, thus, the past states.

Table 19: Hypothetical user schedule

Activity	Time ($t_d - t_a$)	Initial location	Destination	Route	Route (segment roads)	Route (intersections)
A_1	8:00 - 8:15	home	kindergarten	R_1	$s_1 \rightarrow s_7 \rightarrow s_3 \rightarrow s_4$	$x_0 \rightarrow x_1 \rightarrow x_2 \rightarrow x_5 \rightarrow x_7$
A_2	8:20 - 9:00	kindergarten	coffee shop	R_2	$s_4 \rightarrow s_3 \rightarrow s_8 \rightarrow s_6$	$x_7 \rightarrow x_5 \rightarrow x_2 \rightarrow x_3 \rightarrow x_4$
A_3	9:30 - 9:50	coffee shop	work	R_3	$s_6 \rightarrow s_8 \rightarrow s_7 \rightarrow s_2$	$x_4 \rightarrow x_3 \rightarrow x_2 \rightarrow x_1 \rightarrow x_6$
A_4	12:30 - 12:55	work	coffee shop	R_4	$s_2 \rightarrow s_7 \rightarrow s_8 \rightarrow s_6$	$x_6 \rightarrow x_1 \rightarrow x_2 \rightarrow x_3 \rightarrow x_4$
A_5	13:25 - 13:50	coffee shop	kindergarten	R_5	$s_6 \rightarrow s_8 \rightarrow s_3 \rightarrow s_4$	$x_4 \rightarrow x_3 \rightarrow x_2 \rightarrow x_5 \rightarrow x_7$
A_6	14:00 - 14:20	kindergarten	home	R_6	$s_4 \rightarrow s_3 \rightarrow s_7 \rightarrow s_1$	$x_7 \rightarrow x_5 \rightarrow x_2 \rightarrow x_1 \rightarrow x_0$
A_7	15:50 - 16:30	home	college	R_7	$s_1 \rightarrow s_7 \rightarrow s_3 \rightarrow s_5$	$x_0 \rightarrow x_1 \rightarrow x_2 \rightarrow x_5 \rightarrow x_8$
A_8	20:00 - 20:20	college	coffee shop	R_8	$s_5 \rightarrow s_3 \rightarrow s_8 \rightarrow s_6$	$x_8 \rightarrow x_5 \rightarrow x_2 \rightarrow x_3 \rightarrow x_4$
A_9	20:40 - 20:50	coffee shop	home	R_9	$s_6 \rightarrow s_8 \rightarrow s_7 \rightarrow s_1$	$x_4 \rightarrow x_3 \rightarrow x_2 \rightarrow x_1 \rightarrow x_0$

The present model assumes that any road segment (i.e., each S_i , in Figure 18) terminates on an intersection (i.e., each X_i , in Figure 18), and each intersection can be a destination (or a *Staying Point* (SP)), i.e., we do not really differentiate the road intermediate intersection from destination. The representation of location and current state, NMSI, is defined by $NMSI = \langle \text{geographical coordinates (lat,lon), direction, speed (x,y), timestamp (arrive time, depart time)} \rangle$. Our main objective with this proposal is to predict the next destination (SP), or a trajectory (i.e., next k destinations) - hidden states, given a set of NMSI - observable states.

Information about a trip sequence, i.e., consecutive NMSI, can be collected periodically or triggered by events. In order to capture all necessary information to better predict the next location, and consequently the trajectory, our model periodically collects NMSI.

In this proposal, the possible destinations (intersections and SP) are assumed as the hidden states and the visited places (i.e, each position where a distinct NMSI was taken), from each vehicle's mobility pattern history, are considered as observable states. The observable states include the timestamps. The Baum-Welch algorithm is used to estimate the model parameters ($\lambda = (A, B, \pi)$), and train the model with existing mobility traces, which as of our assumptions all vehicles are willing to share.

Considering the characterization of HMM, presented earlier, each NMSI is an observation. Each NMSI is encoded as a specific vocabulary (in HMM nomenclature, an element of $V = v_1, v_2, \dots, v_M$), and a specific sequence of registered NMSI compose the observation sequence ($O = O_1, O_2, \dots, O_T$, in HMM nomenclature). Encoding both the spatial and temporal dimensions allows to, for instance, be able to predict a trajectory from Coffee shop (x_4) to Home (x_0), given only the observed sequence (NMSI), along the segment S_6 , which otherwise could be very difficulty as we have at least three destinations

using three similar segments, and intersections ($x_4 \rightarrow x_3 \rightarrow x_2 \rightarrow$), as shown in the example earlier in this section. The Viterbi algorithm is used to predict the hidden state sequence, that is, given the model a set of observations $O = O_1, O_2, \dots, O_T$, which are a set of taken NMSI, the next k destination will be predicted. The proposed LTMP is presented in Algorithm 4.6.

Algorithm 4.6: The LTMP algorithm, based on HMM

Input : nmsi, currentTime, numLineRead
Output: tuple (predictedPosX, predictedPosY)

```

1 timestamp  $\leftarrow$  currentTime();
2 currentX  $\leftarrow$  getLatPosition(nmsi);
3 currentY  $\leftarrow$  getLonPosition(nmsi);
4 currentVX  $\leftarrow$  getXSpeed(nmsi);
5 currentVY  $\leftarrow$  getYSpeed(nmsi);

6 [px, py, vx, vy]  $\leftarrow$  getMobilityHistoryValues();
7 states.emplace_back(State{px, py, vx, vy});
8 observations.emplace_back(Observation{nmsi});

9 // Training and predicting;
10 hmm  $\leftarrow$  HMM(states, observations);
11 hmm.train(N); // N Baum-Welch iterations, for training the model. We used N = 3;
12 predictedState  $\leftarrow$  hmm.predictNextPosition( State{currentX, currentY, currentVX, currentVY}, Observation{currentVX, currentVY,
    timestamp} );
13 return tuple(predictedPos.x, predictedPos.y);
```

4.7 Summary

This chapter presented the design overview of the proposed model, including the justification of all our design choices. The model is designed having in mind the assumptions presented in Section 1.4. It is designed at the data and control plane of the network layer.

First, we detailed the design of CMAF which after including mobility information on all packets, takes advantage of overheard packets in the wireless channel and use the extracted mobility information to feed the proposed mobility prediction algorithm - STMP. The STMP is used to update the geo-location of nodes in the NT. The referred procedure takes place either proactively to update and manage NT, and reactively whenever a new next-hop is selected. In this first step, CMAF does not use explicitly FIB to decide on the next-hop.

Secondly, we developed MARV, which also resort to a mobility prediction algorithm - LTMP, this time to manage FIB. The LTMP is proactively used to track and update the mobility information of nodes registered as next-hops in FIB. MARV also use information in NT to update and remove FIB entries that are no longer registered as neighbors. After the deployment of MARV, CMAF resort entirely on FIB to decide on the next-hop. The implementation of the model is presented next in Chapter 5.

Implementation of the Proposed Model

Chapter 4 presented the design overview of the models proposed by the present thesis. This chapter focus on presenting the practical implementation of the models. Before delving into the implementation itself, the chapter starts by describing the main technology used for implementing and simulating the models. Hence, it presents and justifies the choices made on selecting the tools used for simulations, including the representation of the main modifications made on the simulator, in order to adapt it to the models. It also highlights challenges faced during the development and simulation.

5.1 Network Simulator

For implementation and simulations, this work uses ndnSIM [171] - a network simulator for NDN architecture. The simulator is chosen based on the fact that the majority of surveyed studies in [232] used ndnSIM and Simulation of Urban MObility (SUMO) [161]. Besides that, works such as [52, 243] compared and largely justified the preference of it over other simulators for NDN. Specifically, [243] grouped the ICN-based architectures in 1) CCN/NDN and 2) all the other ICN-based architectures. Then, compared all simulators/emulators used by these two groups for accuracy, consistency, scalability and fidelity. It concluded that CCN/NDN-based simulators are better. Moreover, among those, ndnSIM presents better results in the chosen metrics, except in scalability. Other evaluation frameworks designed specifically for NDN and not included in the mentioned studies are, miniNDN (for medium scale networks) and NDN Testbed (for large scale wired network). Along the development of this thesis the NDN Testbed was used for testing the impact of scalability and size of CSs in a structured network scenario [234].

ndnSIM is based and leverages Network Simulator 3 (NS-3) [188] specifically to: a) create the simulation topology and specify the respective parameters, b) simulate the link layer protocol model, c) simulate the communication between different NDN nodes, and d) record the simulation events [171]. ndnSIM is implemented as a network layer protocol model, and can work on top of any link layer protocol model [184]. It is an open-source project, modularly developed using C++ programming language. Some of its interfaces are developed using Python. As of version 2.0, the simulator started integrating the NFD [6] code and its supporting libraries - ndn-cxx [203]. One of the main benefits of the integration is the

possibility of simulating codes developed for ndnSIM, directly in NFD and vice-versa [171]. NFD is the real implementation of NDN architecture and can be deployed to run on simulated or on real environments.

The structure of the ndnSIM environment, Figure 19, is mainly composed by the: 1) NS-3 framework, 2) NFD module, 3) ndn-cxx library, 4) NDN simulation layer, 5) specific applications, 6) real applications, and 7) simulation scenarios.

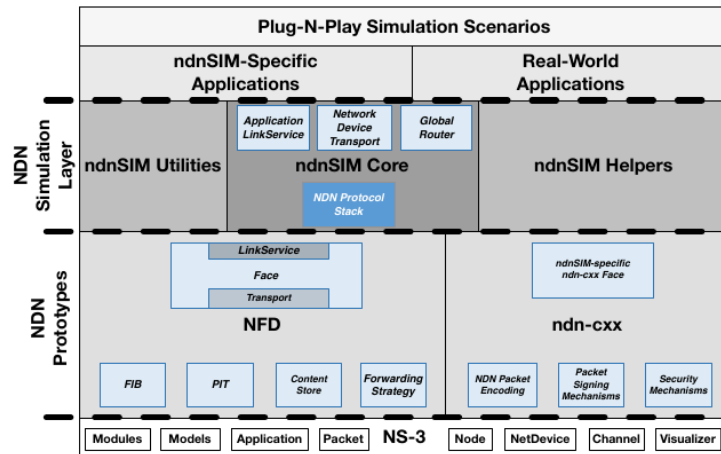


Figure 19: Structure of ndnSIM [171]

ndnSIM includes some specific applications that are used to generate and manage Interest/Data packets. Some of the main ones are presented below [171]:

- Consumer, generates traffic of Interest with a pre-defined pattern (e.g., constant frequency, constant average rate with uniformly or randomly distributed interval between Interest);
- ConsumerZipfMandelbrot, generates traffic from popular Interest, which follows a Zipf-Mandelbrot distribution [51];
- Producer, responds to each Interest with the corresponding Data.

and it integrates the following forwarding strategies, which are implemented in the NFD [6]:

- *Best route*, forwards Interest through Face with lower cost of routing;
- *Multicast*, forwards Interest to all Faces indicated in the corresponding FIB entry;
- *ncc*, forwards packets to manually configured or registered Faces by local applications [269];
- *Client-control*, forwards each Interest through the Faces indicated by the consumer;
- *Access route* (or *access strategy*), forwards Interest based on local prefixes;
- *Adaptive SRTT-based Forwarding* (ASF), forwards Interest using routes with less RTT. Periodically it makes inquiries to record the routes with lowest RTT.

- *self-learning*, wired-network-based, uses broadcast for learning new paths;

This section presents not all the ndnSIM's components, but only the main applications and forwarding strategies given that this work produces mainly new strategies (forwarding - Section 4.3 and routing - Section 4.4) and applications.

ndnSIM requires other tools for creating complex road traffic, for simulations. Next section presents the selected traffic simulator.

5.2 Road Traffic Simulator

This work studies and develops models in VANET environment. Knowing the position and movement of nodes is crucial for VANET. In NS-3 position is represented on a 3D Cartesian (x,y,z) coordinate system, where `<ns3::MobilityHelper class>` methods combine a *mobility model* and *position allocators*. *Position allocators* configure initial position of nodes when the simulation starts and the *mobility models* describe how nodes move during the simulation. Existing *mobility models* include [188] i) Random Way Point - which specify the location for a given time (time-position), ii) Constant position, iii) Constant velocity, iv) Constant acceleration. These models are mainly used for simulating mobility for MANET or simple VANET scenario testing, and are not capable of generating complex vehicle maneuvers including accelerating, decelerating, and traffic jams [30]. Hence, not suited for a more advanced VANET simulation. For more complex scenarios, NS-3 has the possibility of incorporating traces externally generated by specific tools designed to generate complex mobility models. Some of the aforementioned tools include SUMO, BonnMotion, TranS, Vens, whose produced mobility models can be imported to NS-3 using helpers such as the *Ns2MobilityHelper*. Any of the aforementioned tools could be used in this work but We chose SUMO [161], as already justified.

SUMO is a microscopic simulator, capable of generating realistic movements of vehicles. For vehicle mobility, SUMO can generate three specific road models 1) Highway - a straight highway where vehicles move in lanes in one or two directions. This model is suitable for rural scenarios, and 2) Road grid - a grid more suitable for urban scenarios, where vehicles generally move in two dimensions, and 3) Additionally, besides the simple grid, it can model a real road map from any existing city, importing road maps from the Open Street Map (OSM) database. SUMO is able to output mobility traces compatible with the ns-2 mobility format, that can be directly imported by NS-3 and used in ndnSIM.

Specifically for this work, SUMO was used to generate two distinct road model: 1) the Manhattan-like road model - Figure 20 and 2) a model based on real road map from city of Porto, Portugal - Figure 21.

The Manhattan-like road network topology was specified to have a 4×3 dimension grid scenario with an area of $1200 \times 1600 \text{ m}^2$. Each street is bidirectional and has 2 lanes for each direction. Vehicles depart randomly and have varying speeds, from 0 m/s to 16.39 m/s. The probability of turning left or right on intersections is fixed as 50% straight, 25% right and 25% left.

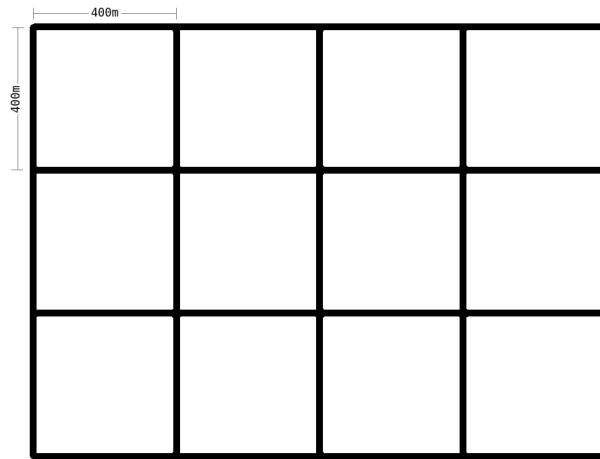


Figure 20: Manhattan-like scenario.

The road topology based on the city of Porto covers an area of $1000 \times 1000 \text{ m}^2$. Vehicles depart randomly and have varying speeds, from 0 m/s to 34.10 m/s. This scenario includes highways and the possibility of having disconnected sections. That is, the city of Porto scenario is more susceptible of having disconnected segments than the Manhattan's. Therefore, the robustness of the proposed CMAF and MARV models for partitioned network are well tested in this scenario.

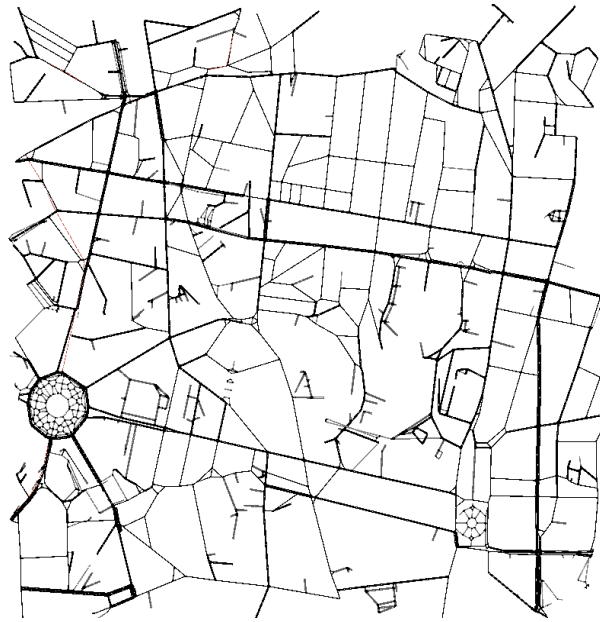


Figure 21: Scenario based on the City of Porto, Portugal.

For either topology, the following are the commands used to produce the aforementioned road traffic networks.

The first command (*Command_1*) is used to generate random trips, with a duration of 260s and with an equidistant departure times and period $p = 0.01$. The command outputs a file (i.e., *routes.rou.xml*) with valid vehicle routes, discarding the disconnected trips. It receives an input file (i.e., *map.net.xml*) containing road network produced by SUMO *netconvert* package, which imports digital road networks

Table 20: SUMO's commands used to produce the trace files with the desirable characteristics

<i>Command_1</i>	<code>python \$SUMO_HOME/tools/randomTrips.py -n map.net.xml -r routes.rou.xml --begin 0 --end 260 -p 0.01 -v</code>
<i>Command_2</i>	<code>sumo -c simple.sumocfg --max-num-vehicles=κ --step-length 0.1 --fcd-output sumoTrace2.xml</code>
<i>Command_3</i>	<code>python processTrace.py</code>
<i>Command_4</i>	<code>python \$SUMO_HOME/tools/traceExporter.py --fcd-input sumoTrace.xml --ns2mobility-output mobility_κ_20230128.tcl</code>

from different sources (e.g., real google maps used for city of Porto or a specific model used to produce the Manhattan-like road topology) and generates the road networks.

The *Command_2* runs SUMO simulation using information provided by *Command_1* (i.e., *routes.rou.xml*) and by *netconvert* (i.e., *map.net.xml*), which are included in the file *simple.sumocfg*. *Command_2* also defines the maximum number of vehicles (i.e., κ) in the simulation. For our case, κ takes the following values 20, 40, 60, 80, 100, and 120 vehicles per simulated scenario. *Command_2* produces the trace file *sumoTrace2.xml* which is, then, processed by *Command_3*.

The files produced by *Command_2* do not ensure a fixed number of κ vehicles for the duration of simulation. It only ensures that at most κ vehicles will be running in the simulation, at a given instant. The *Command_3* is a python script written to reinsert some vehicles that are leaving the simulation, in order to maintain the number of vehicles in κ , all the time.

The fourth and last command (*Command_4*) is used to export the trace file, produced by *Command_3* (i.e., *sumoTrace.xml* from *sumoTrace2.xml*), to NS-2 mobility file, which is then imported into ndnSIM using the `<ns3::Ns2MobilityHelper class>` in `</ns-3/mobility/helper.hpp>` file.

5.3 Technical specifications

The tools presented in Section 5.1 and Section 5.2 were installed on a laptop with the following specifications.

5.3.1 Hardware

The simulations were conducted on a laptop with the following hardware specifications:

- Processor: Intel Core i7-9750H, 6-core (12 logic processors), @ 800/4000 MHz;
- RAM: available: 31.26 GiB;
- Swap: 50 GiB;
- Storage: 970.87 GiB, SSD.

5.3.2 Software

Besides the tools presented in Section 5.1 and Section 5.2, additional software is used, as presented below.

- Operating System: Gentoo Linux, Gentoo Base System release 2.13, with kernel: 6.1.12-gentoo, with GCC-12.3.1;
- Simulation framework: ndnSIM 2.8 (built on NS-3.30.1), as presented in Section 5.1;
- Programming Language: C++ and Python (for automatizing the simulations and for data extraction and analysis);
- Required Libraries: Boost 1.74 (locally installed for ndnSIM, due to incompatibility with system-wide libraries), NS-3 modules.

5.4 System Implementation

The initial procedure for implementing the proposed model in ndnSIM was inspired by [133], which described the methodology for implementing MANET in ndnSIM.

Figure 22 shows the implementation of NDN over WiFi, including the MAC layer managed by NS-3. As the figure depicts, the `<ns3::WifiNetDevice class>` is the highest one that manages the WiFi processing at the ndnSIM level.

In ndnSIM, when an Interest or Data packet is sent, the forwarding `<nfd::fw::Strategy class>` calls the corresponding `<SendInterest()>` or `<SendData()>` methods in the `<ndn::face::Face class>`. These methods, in turn, invoke `<sendPacket()>` method in `<nfd::face::LinkService class>`. The `<sendPacket()>` method invokes `<doSend()>` method in `<ns3::ndn::NetDeviceTransport class>`. The `<doSend()>` calls the `<Send()>` method in the `<ns3::NetDevice class>`, which subsequently calls the `<Send()>` method in the underlying `<ns3::WifiNetDevice class>`. At this point, the packet is handed over to the MAC layer, where the destination MAC address is set. Currently, it defaults to the broadcast address (ff:ff:ff:ff:ff:ff).

Similarly, when an Interest or Data packet is received, the `<ns3::WifiNetDevice class>` invokes the `<ForwardUp()>` method in the `<ns3::NetDevice class>`. The `<ns3::NetDevice class>` forwards the packet to the `<ns3::ndn::NetDeviceTransport class>`, which then calls the `<Receive()>` method in the `<nfd::face::Face class>` through the `<ReceiveFromNetDevice()>` method. From the `<ns3::ndn::NetDeviceTransport class>`, the MAC address is ignored and the subsequent classes processing the packet do not have a means of accessing this value.

To enable the forwarding `<nfd::fw::Strategy class>` to manage MAC address, and be able to process the packets as proposed in Chapter 4, we modified the `<WifiHelper::Install()>` method in `</ns-3/wifi/helper.hpp>`, allowing it to create a file in the disk. The created file ("NodeIdMacMapping") contains in each line, the pair: `<"00:00:00:00:00:01" - 0>`, which is a mapping of the node MAC address and the corresponding node ID. This file is re-created each time the simulation is started and nodes are created. When the simulation is running, the forwarding strategy in each node can access the written file and extract the MAC address of a node, given its unique ID in the network. Additional modifications, which in fact include the MAC address already registered in this description, are presented in next sections.

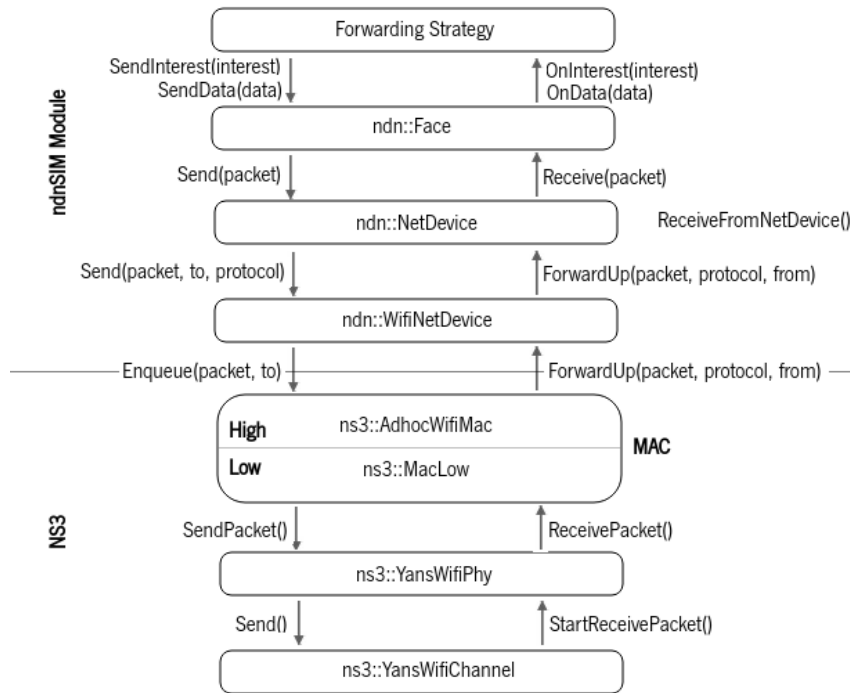


Figure 22: Implementation of the ndnSIM, and the details for packet processing in WiFi.

5.4.1 Main NDN Structures Enhancements

As already referred, the dissemination model of the wireless channel can be explored to reduce the need for a purposely initiated beacon broadcast for discovering neighbors, and to learn about new content sources. In order to take advantage of the overheard packets, all NDN packets in this work, are extended to carry additional (optional) mobility information - the NMSI - which includes: a) the ID of the sender node, as justified in Section 4.4; b) its current speed (from the Global Navigation Satellite System measurements); c) its geographical position; d) the hop count; and e) a timestamp. The NMSI is included in each packet, but its use is optional in the sense that whenever necessary, the model can fallback to the normal operational mode of NDN, flooding the network to discover new neighbors and new content sources. NMSI is added as a tag to Named Data Networking Link Protocol (NDNLP) packet header. When a packet is received, its corresponding NMSI refers to the mobility status of the last node from where the packet was received. The procedure for registering the NMSI and other modification are presented in next sections.

5.4.1.1 Registering NMSI

To implement this modification, a new class - `<ndn::util::Nmsi class>` in `</ndn-cxx/util/nmsi.hpp>` - was created. New fields, from the `<ndn::util::Nmsi class>` and the new added field to Interest and Data packets must be codified for the NDNLP' encoding procedure. This process takes place at the `ndn-cxx` libraries level. Specifically, the following values are added to the file `<ndn-cxx/encoding/tlv.hpp>` and to `<ndn-cxx/encoding/tlv-nfd.hpp>` in order to enable NFD's management to know the new fields. Table 21

shows the coded values.

Table 21: Codified values for NMSI

Modified file	<i>origID</i>	<i>destFinalID</i>	<i>Hops</i>	<i>Px</i>	<i>Py</i>	<i>Vx</i>	<i>Vy</i>	<i>Timestamp</i>	<i>EndpointId</i>
<i>tlv.hpp</i>	38	39	42	40	41	43	44	–	–
<i>tlv-nfd.hpp</i>	115	116	118	119	120	121	122	117	113

The NDNLP requires the registration of the new tags, for the newly created classes. The procedure is performed as follows. As the *GeoTag* - which includes node geographical position in 3D Cartesian (x,y,z) coordinate system, was already declared in `</ndn-cxx/lp/geo-tag.hpp>`, new class for *SpeedTag* - which includes node speed in 3D Cartesian (x,y,z) coordinate system, is declared in `</ndn-cxx/lp/speed-tag.hpp>`. The Type-Length-Value (TLV)-TYPE numbers are declared in `</ndn-cxx/lp/tlv.hpp>` and, as the *GeoTag* numbers were already defined as *GeoTag* = 85, *GeoTagPos* = 85, the newly defined values are presented in Table 22. The tags fields are declared in `</ndn-cxx/lp/fields.hpp>` and their classes registered in `</ndn-cxx/lp/tags.hpp>` as *GeoTag*, *SpeedTag*, *fromIdTag*, *destIdTag*, *timestampTag*. Getters and setters for node geo-position and node speed attributes are declared in separate classes - `<ndn::util::GeoPosition class>` and `<ndn::util::NodeSpeed class>` - from `</ndn-cxx/util/geo-position.hpp>` and `</ndn-cxx/util/node-speed.hpp>`, respectively.

Table 22: TLV-TYPE numbers for new tags

Modified file	<i>SpeedTag</i>	<i>SpeedTagPos</i>	<i>fromId</i>	<i>destId</i>	<i>timestamp</i>
<i>tlv.hpp</i>	87	87	92	93	94

It worth clarifying that NMSI information is added to the packet right before it is sent out from the sending node. The procedure takes place at the `<nfd::face::GenericLinkService class>` from `</NFD/daemon/face/generic-link-service.hpp>` file. It is the same class that extracts it and appends it to Interest and Data packets, before they are forwarded up to the network layer.

5.4.1.2 Procedure for Modifying Interest and Data packets

The `<ndn::Interest class>` and `<ndn::Data class>`, for Interest and Data respectively, are declared in `</ndn-cxx/Interest.hpp>` and `</ndn-cxx/Data.hpp>`, respectively. Both classes are extended with two additional attributes (i.e., *origID* and *destFinalID*, as string values for storing MAC Address), including their respective setters and getters methods. The TLV-TYPE numbers for Interest and Data wire-encoding/decoding are as presented in Table 21. The mentioned procedure is sufficient for adding the new attributes to the packets.

5.4.1.3 Procedure for Modifying PIT

PIT entry - `<nfd::pit::Entry class>` - represents either a pending or a recently satisfied Interest. It is identified by an Interest [6]. When the requested content is received, a PIT lookup takes place in order to find the Face to forward the content to. As aforementioned, for a wired network, a Face unambiguously identifies the next-hop. The Face-based communication mechanism does not work well for wireless channel, given that all packets are forwarded-out from the same Face. To overcome this issue and be able to identify the possible next-hop nodes for Data delivery on downstream, the ID of the node from which the Interest was received (*fromId*) is included in the corresponding PIT entry alongside the Face, see Figure 23. As mentioned, a good candidate for node identification is the MAC address of the Faces (i.e., net devices) installed on these nodes. The node identified by *fromId* is the first choice for Data forwarding. However, as Data packet includes the ID of the original requester - the consumer, (as explained in Section 4.3.3), the current node knows where the packet should be sent to. Thus, besides the node identified by *fromId*, whenever an intermediate good candidate for relay exists, it is selected to forward the content. In this case, the next-hop relay which is included in the packet is updated accordingly.

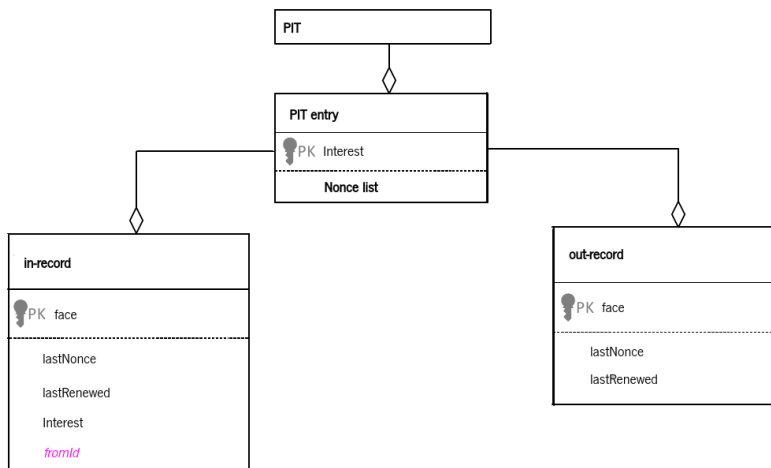


Figure 23: Structure of PIT with the additional field - *fromId*.

The `<nfd::pit::Pit class>` in `</NFD/daemon/table/pit.hpp>`, `<nfd::pit::FaceRecord class>` in `<pit-face-record.hpp>`, and `<nfd::pit::Entry class>` in `</NFD/daemon/table/pit-entry.hpp>` are extended to include the new attribute and their respective additional polymorphic methods (i.e., `<getInRecord()>`, `<insertOrUpdateInRecord()>`, `<deleteInRecord()>`, `<getOutRecord()>`, `<insertOrUpdateOutRecord()>`, `<deleteOutRecord()>`) for handling it. The PIT entry to remove, for instance, should now be identified not only by its Face but also by its *fromId* attribute.

5.4.1.4 Procedure for Modifying FIB

FIB is also one of the three main structure used by NDN for managing the forwarding Faces.

The `<FibHelper::AddRoute()>` method from `<ns3::ndn::FibHelper class>` in `</ndnSIM/helper/fib-helper.hpp>` is initially invoked by the producer applications when registering their own FIB routes for the

prefixes they have the Data. Then, MARV uses the method (and the `<FibHelper::RemoveRoute()>`) for managing FIB in every node. Therefore, the class is extended accordingly to include new polymorphic methods (e.g., `<FibHelper::AddRoute()>`/`<FibHelper::RemoveRoute()>`) with additional parameters related to the new attributes from the `<ndn::util::Nmsi class>`. Moreover, the referred modification implies updating `<StackHelper::createAndRegisterFace()>` and add a new polymorphic `<StackHelper::SetDefaultRoutes()>` method, with the new attributes in `</ndnSIM/helper/ndn-stack-helper.hpp>`.

Figure 24 depicts the FIB structure and the modification to include the new fields for NMSI attributes.

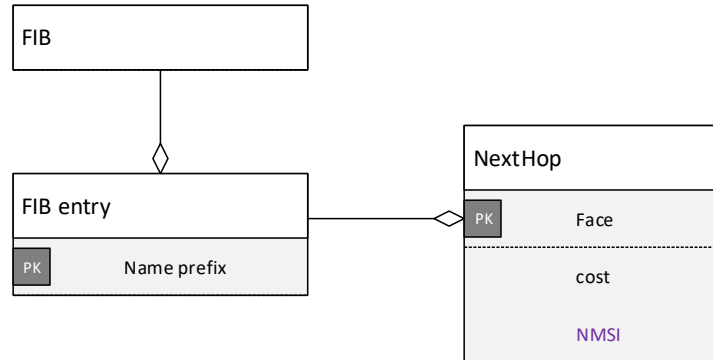


Figure 24: Structure of FIB with additional fields for NMSI for each next-hop.

5.4.2 Procedure for Modifying the Forwarder

To accommodate the specific aspects of the designed model, the `<nfd::Forwarder class>` was also modified. The modifications are presented next.

- In order to enable PIT to record not only the Face but also the *fromId* value, the `<nfd::pit::Entry class>` in `</NFD/daemon/table/pit-entry.hpp>` file was modified as explained in Section 5.4.1.3. To handle the referred modification, all methods in the `<nfd::Forwarder class>` that invoke any of the mentioned modified methods from `<nfd::pit::Entry class>` are also modified accordingly.
- As aforementioned, the model caches all Data, including the unsolicited, except all active-safety- and some efficiency-related content. The selection of which Data to cache, is also performed in the `<nfd::Forwarder class>`.
- In the current form of MARV, the timer-based logic for managing FIB is also coded in the `<nfd::Forwarder class>`.

5.4.3 Procedure for Adding new Forwarding Strategy

The NFD guide [6] describes the methodology one should use to develop and install a new forwarding strategy in ndnSIM. This work adopts the methodology described there. The design of the strategy is described in Chapter 4.3.

We chose an existing forwarding strategy from `<NFD/daemon/fw>` and edited it accordingly to accommodate the design presented in the previous chapter. The proposed forwarding Strategy is implemented by the `<nfd::fw::MobilityAwareStrategy class>` in `<NFD/daemon/fw/mobility-aware-strategy.hpp>`, and inherits from `<nfd::fw::Strategy class>`. Two methods are of interest when implementing a new strategy: 1) `<afterReceiveInterest()>` and `<afterReceiveData()>`, invoked by `<onIncomingInterest()>` and by `<onIncomingData()>` methods from the `<nfd::Forwarding class>` in `<NFD/daemon/fw/forwarder.hpp>`, when an Interest or a Data packet is received, respectively. Both methods are responsible for deciding whether the received Interest or Data packet should be or not forwarded further.

When `<afterReceiveInterest()>` or `afterReceiveData()` method is invoked, the first action it takes is to extract NMSI from the received packet, and use the information to update NT and NTRsu if applicable. Then, if the packet is active-safety- or efficiency-related it is immediately broadcast. Otherwise, it is processed as presented in Section 4.3.4 or Section 4.3.3, respectively.

After CMAF has decided to forward the packet, it appends the newly chosen next-hop as a tag to the packet (i.e., `<packet.setTag<lp::destIdTag>(nake_shared<lp::destIdTag>(selected_next_hop))>`) prior to sending it out. The new NMSI information is also registered at this stage. The `destIdTag` is the key value for enabling unicast communications for CMAF. When a packet is received, as explained earlier, after extracting and using NMSI to update NT, the `destIdTag` is extracted (`<getTag<lp::destIdTag>()>`) and if the value it carries corresponds to the ID of the node processing it, then the packet has arrived to the destined node and is eligible for further processing. Otherwise, it is discarded, as the important information has already been extracted. The model uses the promiscuous mode - which can be set in `</ndnSIM/model/ndn-net-device-transport.hpp>`, hence, all received packets at the `<ns3::WiFiNetDevice class>` are forwarded-up to the network layer - to both, the forwarding and the routing planes.

The strategy includes three additional tables: 1) the CCT, 2) the NTRsu and 3) the NT. All are defined in `</NFD/daemon/fw/mobility-aware-strategy.hpp>`.

1. the CCT is used in each node to catalog the cached Data of all nodes with which the node has interacted before. Each CCT entry is a tuple composed by the ID of the node from which a packet has been received, and the BF that codifies the catalog of the contents of that node's CS. It is defined as `std::list<std::tuple<uint64_t, ::ndn::util::BloomFilter>>`. If a BF is received, the CMAF searches CCT. If the ID of the node from which the packet have been received is present in the CCT, then the received BF is updated accordingly. Otherwise, it is a new node and, therefore, a new entry is created.
2. the NTRsu - used to store the list of neighbor RSUs. The IDs in this list, which correspond to the existing RSU, are persistently stored. That is, the IDs are currently never removed.
3. the NT is used to store the list of mobile neighbor nodes. In fact, all RSUs are also added to this table but, differently from the NTRsu, when a predefined time elapses without any contact from

the RSU, as it happens with any mobile node, its ID is removed from the NT. The insertion of a new neighbor and the maintenance of the NT is described in the Section 4.3.2.

5.4.4 Procedure for Adding new Routing protocol

Differently from developing and installing a new forwarding strategy, a procedure clearly described in the NFD guide [6], the methodology for developing and installing a new routing protocol is not provided. In the current form, MARV is developed in the same class as CMAF, except the timer-based logic for managing FIB which is coded in the `<nfd::Forwarder class>`. However, it is desirable to have the routing and the forwarding strategy separated from each other. We plan this for future work. It is developed as a method that uses STMP and resort to timers to control the periodicity of updating FIB. It is an endless process that starts running as soon as the first Data packets is received and a corresponding Interest corresponds to an entry in PIT. That is, when the first Interest is satisfied. The routing procedure is described in Section 4.4.

5.4.5 Additional Applications and Classes Included in the Simulator

Several other applications/classes were designed and deployed besides the main ones presented in the former sections, some of which are presented in this section.

- Two mobility prediction algorithms were designed for tracking the mobility of nodes: STMP and LTMP. To implement the algorithms, two classes - the `<ndn::util::StmpKF class>` in the file `</ndn-cxx/util/stmp-kf.hpp>` and its extended version in the `</ndn-cxx/util/extended-kalman-filter.hpp>` and the `<ndn::util::LtmpHMM class>` in the file `</ndn-cxx/util/ltmp-hmm.hpp>` - are created for each algorithm, respectively. The created classes are used by the forwarding strategy and the routing protocol, as presented in Section 4.3 and Section 4.4. The LTMP algorithm is only deployed in the existing RSUs.
- A BF-based algorithm for cataloging the neighbor CS is created and used as explained in Section 4.3.2.2. The respective class - `<ndn::util::BloomFilter class>` - is created in the file `</ndn-cxx/util/bloom-filter.hpp>`.
- The `<nfd::face::GenericLinkService class>` in `</NFD/daemon/face/generic-link-service.hpp>` has been one of the target for modifying the ndnSIM in order to adapt it for VANET. We also modified the class to enable inclusion of NMSI and this level and the corresponding encoding and decoding of its related field tags (i.e., `<lp::SpeedTag>`, `<lp::destIdTag>`, `<lp::fromIdField>`, `<lp::timestampField>`).

For this work, additional applications were developed and corresponding classes stored in the folder `</ndnSIM/apps/>`. The applications are presented next:

- Consumers, inherit from the general Consumer application, defined by the `<ns3::ndn::Consumer class>` in `</ndnSIM/apps/ndn-consumer.hpp>`

- ConsumerCromo2 - adapted to support more general features, related to the designed forwarding strategy and routing protocol;
 - ConsumerCromo2Beacon - adapted specifically to provide beaconing at the RSUs;
 - ConsumerCromo2Emergency - adapted specifically to request active-safety and efficiency-related information;
 - ConsumerCromo2File - adapted specifically to request real content - a real file from local storage. This was done to test the model for disseminating real content. We tested the model with a real file in the disk - a picture in *.png* format. The corresponding producer, see below, responded adequately but the overall performance was poor, due to bad design of the window logic for retrieving the sequential file chunks.
- Producers, inherit from the general *Producer* application, defined by the `<ns3::ndn::Producer class>` in `</ndnSIM/apps/ndn-producer.hpp>`
 - ProducerCromo2 - adapted to respond to ConsumerCromo2 application;
 - ProducerCromo2Beacon - adapted specifically to respond to ConsumerCromo2Beacon application;
 - ProducerCromo2Emergency - adapted specifically to respond to ConsumerCromo2Emergency application;
 - ProducerCromo2File - responds to ConsumerCromo2File application with the requested file.

5.5 Main Challenges on Implementing the Model

- To implement the proposed model some internal components of the chosen simulator, ndnSIM, required modifications in order to adapt it to specific requirements of the model. Moreover, in order to be able to correctly extract logs and related information for debugging, the simulator has to be configured in *debug* mode - `<$. /waf configure debug>`, as suggested by its documentations. It turns out that simulations, with the simulator configured in this mode, takes too much time to finish. After several simulations that took months to finish, we reconfigured the simulator in a *optimized* mode `<$. /waf configure optimized>`. This modification reduced the simulation times of the remaining simulations by about 10 times.
- To further reduce the simulation time, the laptop used for simulations was configured with a temporary file system in RAM, in order to allocate the running simulation including all its auxiliary files for logs and metric extraction directly on the RAM. That is, all required simulation outputs were being periodically dumped in files residing in RAM. This configuration was used and worked well for simulation of CMAF. As already referred routing in a very dynamic network imposes several

challenges. One of the challenges is the maintenance of updated routes in routing tables. Thus, although we already knew that route maintenance implies increased demand in terms of computational resources, we had no clear idea of how demanding this process could be. The simulations were performed in a 6-core laptop capably of executing a total of 12 threads (logical processors) simultaneously, as described in Section 6.2. ndnSIM does not support the use of multiprocessors by default. Its documentation describes the procedure of speeding up the simulations by resorting to parallel execution using Message Passing Interface (MPI) support and the distributed simulator class provided by NS-3, and allocating each network node to a specific logical processor. The mentioned procedure is a bit of complex and not feasible for a network with varying network nodes, where nodes can be inserted and removed on demand. Instead of allocating each node to a specific logical processor, we allocated each simulation to a specific logical processor. So, theoretically we could run 12 simulations simultaneously. When simulating MARV we noted that all existing RAM (i.e., 32GB RAM) were completely allocated and not enough to run 12 simulations in parallel. When reducing the number of simulation, for 3 for instance, each would use around 30% of existing RAM. This would result in a practically stalled system, sometimes taking more than 20 days to finish only 3 simulations, for scenarios with 100 and 120 nodes. Therefore, for MARV the system's simulation environment was reconfigured in order to have all the auxiliary files, for output collecting, in the disk instead of RAM.

- Initially, STMP was designed to run at the forwarding plane and LTMP at the routing plane. Initial simulations showed that running LTMP in all nodes resulted in high Interest Satisfaction Delay (ISD), due to processing overhead. In fact, we were forcing the nodes to run two mobility prediction algorithms. Hence, we decided to run STMP in all nodes at the two planes (i.e., the forwarding and the routing) and leave the LTMP for RSUs, in V2I mode.

5.6 Summary

This section presented the implementation of the proposed model, whose design was presented in the Chapter 4. The chapter described the procedure for allowing the forwarding `<nfd::fw::Strategy class>` to access the MAC address in each node. From there, it detailed the procedure: a) for registering NMSI; b) for modifying the Interest and the Data packet; c) for modifying PIT and FIB; d) for modifying the `<nfd::Forwarder class>`; e) for adding new forwarding strategy and new routing protocol; f) for including additional applications and classes in the simulator; and g) finally presented the main challenges encountered on implementing the model in ndnSIM.

Table 23 summarizes the main classes developed by this work, including their corresponding created files and the roles of each class. The code related to MARV is written in forwarding `<nfd::fw::Strategy class>` and in the `<nfd::Forwarder class>`.

Table 23: Summary of main implemented new classes.

Class	Related files	Role
BloomFilter	bloom-filter{hpp}	Defines and implements the Bloom filter, used to catalog the CS contents.
nmsi	nmsi.hpp	Declares the NMSI, appended onto each transmitted packet.
StmpKF	stmp-kf.{hpp,cpp}	Defines and implements the KF's. First implementation of STMP algorithm.
extendedKF	extended-kalman-filter{hpp,cpp}	Defines and implements the STMP algorithm. Adopted algorithm.
LtmpHMM	ltmp-hmm{hpp,cpp}	Defines and implements the LTMP algorithm.
Tools	tools{hpp,cpp}	Defines and implements various common helper functions, required by other classes.
NodeSpeed	node-speed.hpp	Represent the node speed.
MobilityAwareStrategy	mobility-aware-strategy.{hpp,cpp}	Derived from the <code><nfd::fw::Strategy class></code> , it defines and implements the CMAF.
NeighborhoodList	neighborhood-list.{hpp,cpp}	Defines the neighbor list. Includes related functions such as the self purging method
Neighbor	neighbor.{hpp,cpp}	Defines the neighbor. Includes proprieties such as the neighbor' status, distance to the current node, and its NMSI.

NDN-based VANET: Performance Evaluation

This section presents the simulation environment and the simulation results. Extensive simulations were performed to assess the performance of the proposed model. The model is evaluated in three distinct modes: (i) V2V without a beacon broadcast; (ii) V2I with a beacon broadcast at a frequency of a beacons each 5s; and (iii) V2I with a beacon broadcast at a frequency of a beacons each 10s. The results of the simulations are compared with the following similar works: (a) V-NDN protocol [97], without limiting the hop count to 5, which worsens the overall performance mainly for dense networks (e.g., network with 100, 120 nodes); (b) mutlicast [6]; and (c) CCLF [65]. Two network topologies (City of Porto and Manhattan-like network) are chosen for simulations as presented in Section 6.2.

6.1 Selected Metrics

We select six metrics for performance assessment and comparison with similar studies. These are the metrics usually used to assess the performance of similar studies.

- Interest Satisfaction Ratio (ISR)—Also dubbed the Packet Delivery Ratio (PDR), this refers to the ratio between the satisfied Interest (or the corresponding received Data) to the total Interest transmitted by the consumer(s). We compute this value at the consumer's application Face.
- Interest Satisfaction Delay (ISD)—Refers to the amount of time taken from the Interest being sent by the consumer to the time the corresponding Data are received (i.e., total RTT). Two values can be analyzed: The last delay, which corresponds to the hiatus for the last Interest transmitted, and the full delay which refers to the delay, including retransmissions. We only consider the full delay.
- Jitter—For real-time applications, such as streaming a video file, delay between receiving a pair of video chunks should remain constant. Variations in the delays on streaming chunks of the the video can compromise the quality of the streaming. Jitter can be measured as the statistical variance in the latency/delay of received Data packets [88];

- Hop count—The requested content may be several nodes away from its holder. In a dynamic environment such as the case of VANET, it is even expected to receive content from a “data mule” node, bringing the content from an even further distance. The hop count refers to the average number of nodes Data packets traverse from the content holder (either a producer or any other intermediate node holding the Data) to the consumer.
- Number of retransmissions—When the Interest lifetime expires before its corresponding Data packets are received by the consumer, the consumer may issue another request for the same Data, should it still want the content. The number of retransmissions refers to the average number of transmissions needed to finally fetch the desired content.
- TO—The total number of all packets (i.e., Interest, Data, and control/management packets such as beacons) in the network. It measures the level of congestion in the network. This value is further normalized by the TO of the basic NDN flooding protocol.

6.2 Simulation Environment

For simulations, we use the network simulator for NDN (ndnSIM) [171] version 2.8, along with SUMO [161] version 1.16.0, used to generate the mobility of the nodes. ndnSIM is a modular simulator, written in C++ programming language, and runs under the NS-3 [188], which is largely used for simulations in VANET [52]. It leverages the NS-3 to create the simulation topology and specify the respective parameters to simulate the link layer protocol model and the communication between nodes, and record the simulation events [171]. The simulator was extended to implement the proposed CMAF and MARV models, as described in previous chapter. Related work proposals were also implemented in order to allow for a comparative analysis.

Two road network topologies are used for simulations, as presented in Section 5.2.

Table 24 presents the protocol parameters, and Table 25 the parameters configured for simulations. Some of the parameters in Table 24 were defined after tweaking them in prior simulations.

Understanding some of the parameters requires explanation, and thus we provide it next.

- Simulations time: Configured to run for 260s. For data extraction, however, we limited the period sample to 250s, ignoring the first 10s used as the warm-up time.
- Number of simulations per scenario: NS-3 can be configured to produce deterministic or random results. In order to be able to compute statistics on a large number of independent runs it is necessary to run the simulations as a sequence of independent trials. We resort to the NS-3 substream capability to produce multiple independent runs of the same simulations [188]. For a fixed seed, we randomly selected 30 values to use as run number. This is done by setting the `NS_GLOBAL_VALUE` as follows, for each simulation run: `<NS_GLOBAL_VALUE="RngRun =  $\tau$ " ./waf -run program-name>` where τ is one of the 30 randomly chosen values.

- Network density per scenario: Although possible, we did not force vehicles to depart at $time = 0.0$. We calibrated the maximum number of vehicles in the scenario so that vehicles could enter and depart from the scenario at their specific time. However, all vehicles (maximum number of vehicles per scenario, e.g., 20 for scenario with 20 nodes) are in the simulation after the first 10 s.
- $t_{NTpurge}$: The time for purging nodes in NT. $t_{NTpurge} = 1900ms$ was used considering the time for traveling 250m (the defined transmission range), with a maximum speed of 20m/s in the urban scenario. With higher $t_{NTpurge}$, the average number of neighbors in the NT is more consistent.

Table 24: Parameters of the protocol.

Configured Parameters	Value
$t_{neighborUpdate}$	600ms
$t_{NTpurge}$	$3 \times t_{neighborUpdate} + 100 \text{ ms} = 1900ms$
f_{RSUBf}	0.2 Beacons/s, 0.1 Beacons/s

Table 25: Parameters for the simulation's environment.

Configured Parameters	Value
Network Topology/Scenario	City of Porto, Manhattan-like scenario
Number of producers	1 (randomly selected)
Number of consumers	1 (randomly selected)
Content request rate	5packets/s
Forwarding strategy	CMAF
Propagation Loss Model	Two Ray Ground
Vehicle speed	Max 16.39m/s Manhattan-like. Max 34.10m/s Porto
Wireless Comm. Standard	IEEE802.11p
Data rate	12Mbps
Channel bandwidth	10MHz
Number of antennas per node	1 (Omni directional)
TxPowerStart	12.0dbm
TxPowerEnd	12.0dbm
TxGain	1.0dbm
RxGain	1.0dbm
Transmission Range	250m
Network density (# nodes)	20, 40, 60, 80, 100, 120
Simulations time	260s
Number of simulations per scenario	30 (With different run number)
Interest lifetime	8s
Retransmission timeout (t_{RTX})	3s

6.3 Simulations results for CMAF

This section presents the simulation results for CMAF, where it works without explicitly resorting to FIB. It has been already mentioned that data plane is able to work without resorting to the control plane due to its stateful capability. CMAF explores the aforementioned fact and works independent from the control plane. In the this stage, FIB is only used by the producer to register entries related to prefixes of Data it provides.

Section 6.3.1 presents simulation results using the city of Porto's road network, and Section 6.3.2 presents simulation results using the Manhattan-like grid. All results are extracted and compared with different numbers of nodes in the network (i.e., 20, 40, 60, 80, 100, and 120 nodes).

Note that the chosen number of nodes are just a representative number for simulation purpose. The number of nodes in a real situation in the City of Porto scenario can be much higher (hundreds) in the chosen area.

6.3.1 Simulations Using the Road Scenario from the City of Porto

As already mentioned, due to the packet size, Data flooding is more problematic than Interest flooding. Thus, managing the Data hop count is crucial. In this section, we present the simulation results for three specific scenario configurations: 1) without limiting the Data hop count, 2) limiting the Data hop count as explained in Section 4.3.3.2, and 3) limiting the Interest and Data hop counts.

6.3.1.1 Simulation Results without Limiting Data Hop Count

Figure 25a shows the ISR for the selected strategies (i.e., CMAF without STMP—flooding, CMAF V2V, V-NDN, CCLF, Multicast, CMAF V2I with 0.2 beacons/s, CMAF V2I with 0.1 beacons/s). Its is important to highlight that CMAF does not manage FIB. The multicast strategy resorts to FIB and sends Interest to all upstreams, based on it [6]. The results presented in Figure 25a should be analyzed with the results presented in Figure 25b that depict the level of TO produced by each protocol. CMAF without STMP performs pure flooding. CMAF V2V uses STMP for predicting the next node location as explained before. The presented results show that CMAF V2V has the same results as the CMAF without STMP with pure flooding but with a reduced TO, corresponding to a gain ranging from 7.5% to 22.5%. It means that with the mobility prediction, forwarding is more efficient and only the chosen nodes perform forwarding. Strategies that purely resort to FIB (e.g., multicast) perform badly in VANET as shown by the multicast result. CMAF V2V outperforms all other compared protocols, and the performance is consistently increasing when the number of nodes increases as expected. This is due to the fact that in a dense network, the content availability is greater than in a sparse network. The content availability can be either from the existence of a large number of CS or the high probability of completing the route to the content sources.

The ISR of CMAF V2I is worse than that of CMAF V2V. This is due to the additional packets (i.e., beacons) issued by the RSU. Increasing the frequency of beacon broadcast (i.e., 0.2 beacons/s) worsens

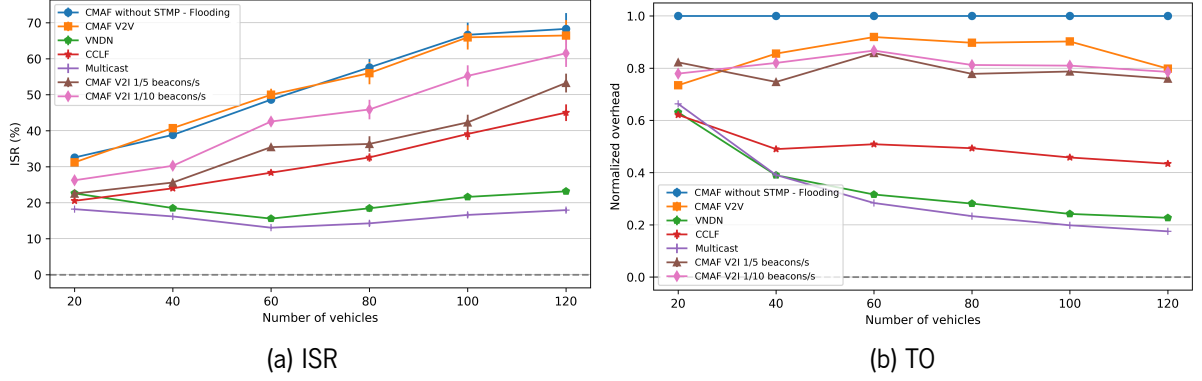


Figure 25: ISR and TO for the selected protocols without Data hop-limit, with CMAF presenting the better performance.

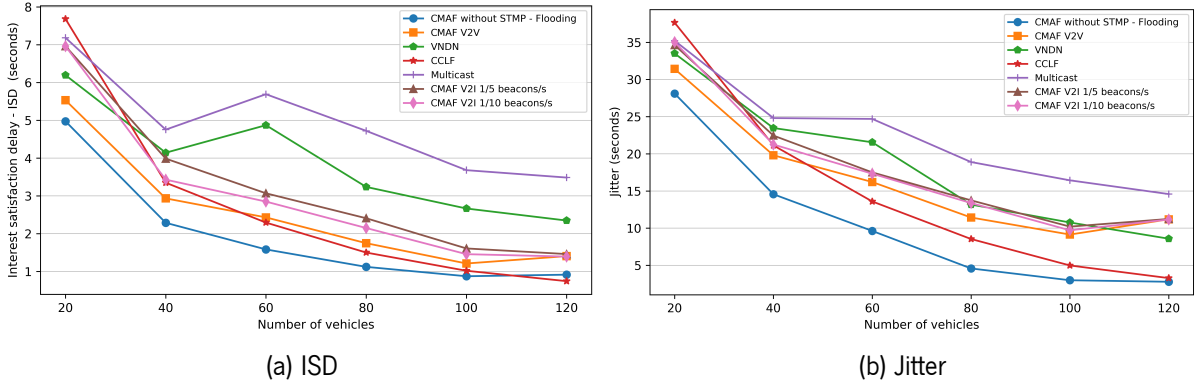


Figure 26: ISD and jitter for the selected protocols, for simulations without limiting Data hop count.

the ISR for the same TO.

The high performance of CMAF V2V is gained at the cost of an increased ISD (maximum of 1 s from the flooding protocol) as presented in Figure 26a. In terms of ISD and jitter (Figure 26b), CCLF performs well mainly for dense networks. This is due to the fact that CCLF presents low TO.

The increased ISD can be justified by conjugating the level of packet retransmissions (Figure 27a) and the packet hop count (Figure 27b). In general, for CMAF, Interests traverse more nodes to fetch the requested content. Consistently, in a dense network, Interests traverse more nodes to fetch the requested content, and the corresponding content is easily returned to the consumer with fewer retransmissions. In a sparse network, nodes are dispersed, and the Interests lifetime may easily expire before they reach the content provider, forcing more retransmission from the consumer. Comparing the scenario with 120 nodes (a dense network), we note that for the same level of packet retransmissions for CMAF and CCLF, the latter presents fewer traversed nodes to fetch content. Therefore, for dense networks, the CCLF ISD is better than that of CMAF. As expected, V-NDN and multicast present the highest ISD, which is justified by their results in terms of packet retransmissions and average hop count.

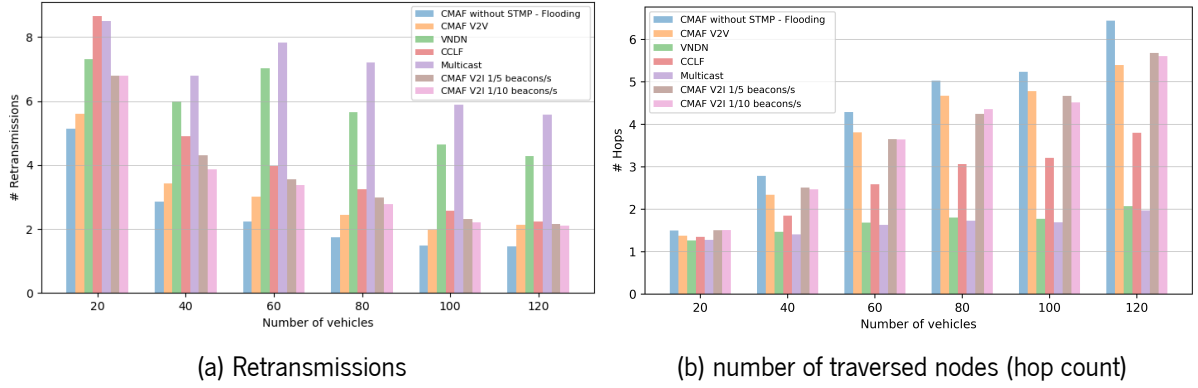


Figure 27: Average packet retransmissions and hop count presented by the protocols.

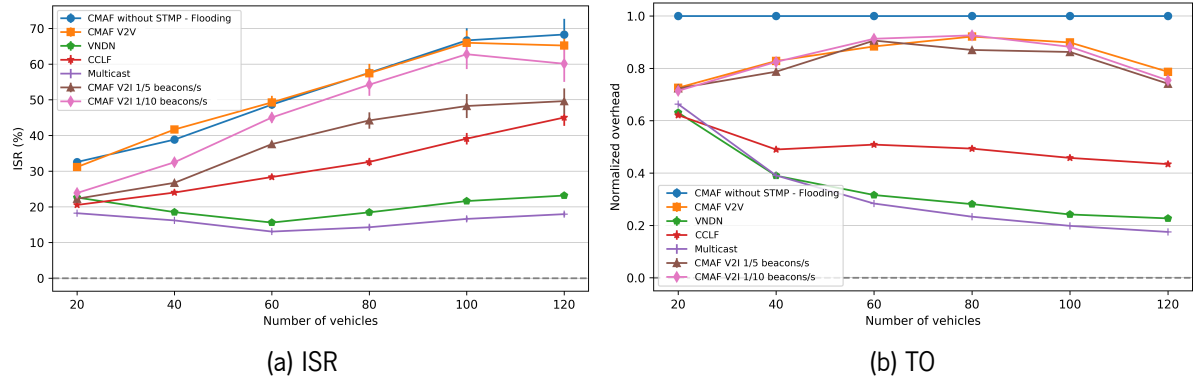


Figure 28: ISR and TO, with limited Data forwarding hop-count.

6.3.1.2 Simulation Results Limiting Data Hop Count

We performed a set of simulations varying the *threshold* used for limiting the Data hop count. If the hop count is limited to the threshold as presented in Section 4.3.3.2 with $\alpha = 14$ and $\beta = 2$, the ISR and TO are as depicted in Figure 28a and 28b, respectively. The results are consistent with the ones presented in the previous section. In terms of ISR, CMAF V2V outperforms all other compared protocols. The performance is consistently increasing when the number of nodes increases as expected. In the present case, however, the network with 120 nodes presents an exception. The ISR of CMAF V2V or V2I for the mentioned scenario (i.e., network with 120 nodes) is relatively lower than the ISR of CMAF—flooding, comparatively with the other scenarios with lesser nodes. It means that although the Interests are allowed and able to traverse many nodes to the corresponding content provider, for dense networks, the chosen *threshold* for Data forwarding is relatively low, causing Data to be discarded early before reaching their destination. This can be observed by comparing the average hop count from Figure 27 and the Figure 32 from Section 6.3.1.3. Only the average hop count of the network with 120 nodes is significantly decreased. For networks with 60, 80, and 100 nodes, for instance, the overall average hop count is even increased with this new configuration (i.e., with limited Data hop count).

From the results, we also highlight that when limiting the Data hop count, the network resources are better managed and, as a result, the scenario (i.e., CMAF V2I) that presents lower performance (see

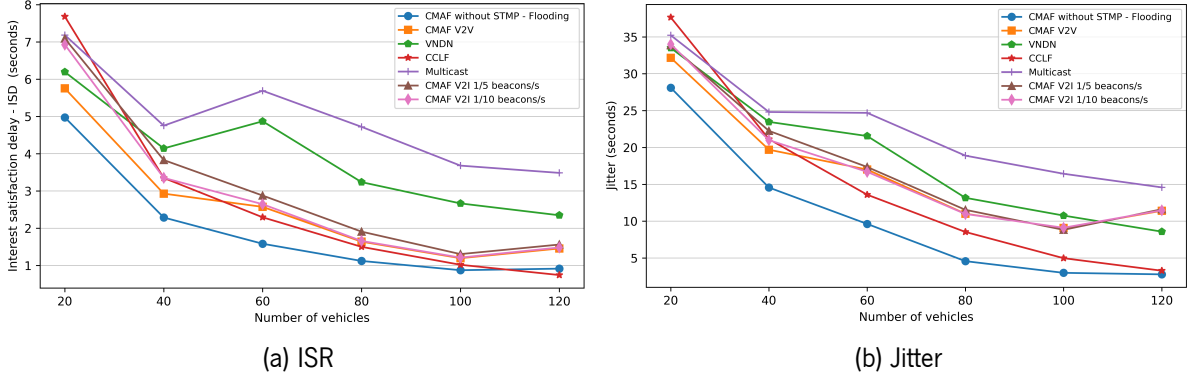


Figure 29: ISD and jitter for the selected protocols, for simulations with limited Data hop-count.

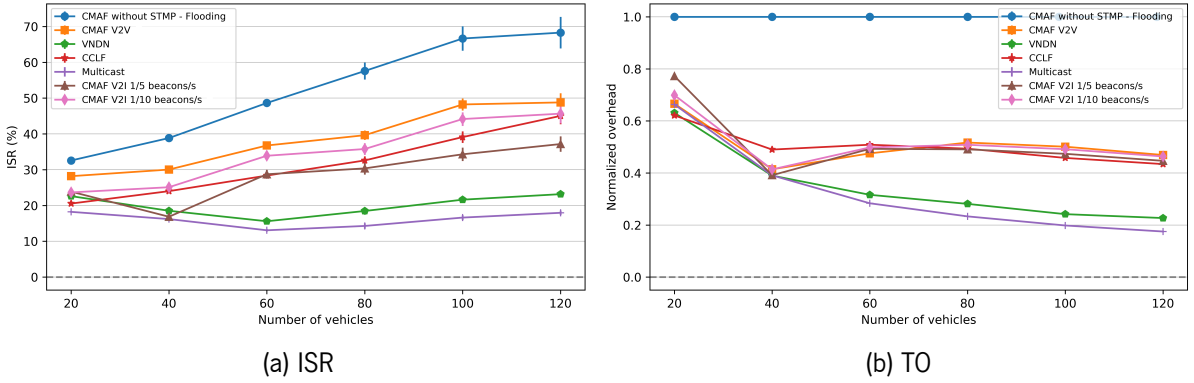


Figure 30: ISR and TO, where Interest and Data forwarding are only allowed for limited hops, in order to have the CMAF TO with the same order of magnitude as that of CCLF.

Figure 25) due to the additional packets (i.e. beacons) now presents improved performance at the cost of an increased ISD and jitter, Figure 29, and with a TO majored by the CMAF V2V.

6.3.1.3 Simulation Results Limiting Interest and Data Hop Count

For the sake of comparison, we further investigate the performance of CMAF with TO as the same order of magnitude as the presented by CCLF. For this matter, both the Data and Interest hop counts are limited. For this simulation, $threshold = \{2, 2, 4, 5, 6, 7\}$ for both Data and Interest packets, respectively, for 20, 40, 60, 80, 100 and 120 nodes. The chosen values are a result of tentative simulation configurations. The results are presented in Figure 30a and 30b for ISR and TO, respectively.

As depicted, CMAF V2V still presents the best performance (i.e., ISR vs. TO) at the cost of an increased ISD and jitter (Figure 31) as expected due to the fewer hops traversed by nodes in CMAF, as shown in Figure 32b. CMAF V2V and V2I also present fewer retransmissions (except the case of the scenario with 120 nodes) than CMAF, as depicted in Figure 32a.

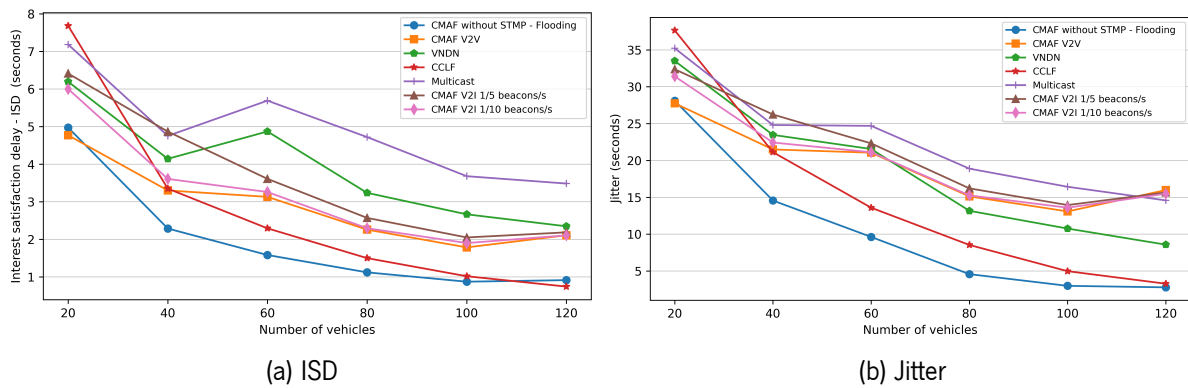


Figure 31: ISD and jitter for the selected protocols, for simulations with limited Interest and Data hop-counts.

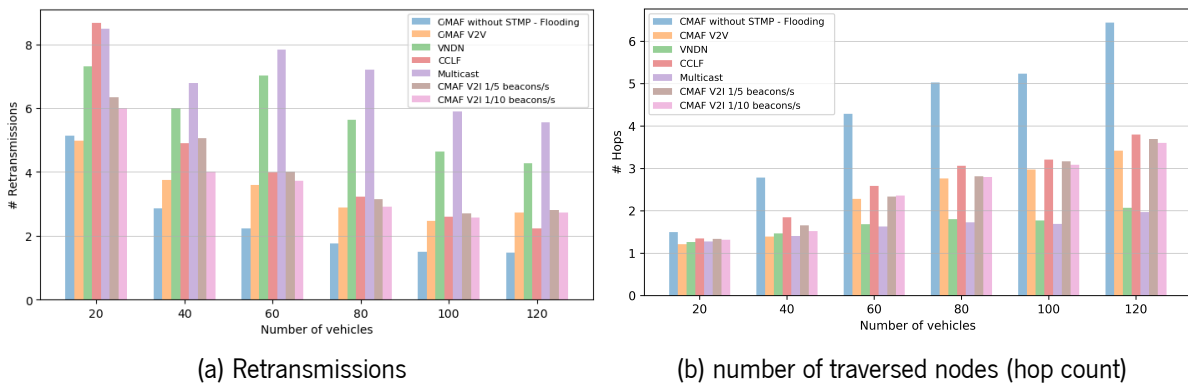


Figure 32: Average packet retransmissions and hop count presented by CMAF, for scenario with TO with the same order of magnitude as that of the CCLF.

6.3.2 Simulations Using a Manhattan-Like Road Network

In the Manhattan-like grid, the nodes are more likely to re-encounter each other during the simulations, than in the city of Porto's road network, where several roads are disconnected and content forwarding may significantly resort to the so-called "data mules". However, the overall results presented in this section are consistent with the ones presented in Section 6.3.1.

Figure 33a and 33b portray, respectively, the ISR and TO of the evaluated protocols. Again, and now with an average gain of 20% in terms of reduced TO compared to the CMAF—flooding, the proposed CMAF presents the best performance among the compared state-of-the-art protocols. The presented performance is relatively high than that presented for the city of Porto's road network due to the fact presented in the previous paragraph.

Figure 34a and 34b present the ISD and jitter, respectively. Again, as presented in the previous sections, in terms of the ISD and jitter, CCLF performs well mainly for dense networks.

CMAF presents fewer retransmissions (Figure 35a) as a result of more traversed nodes (Figure 35b) than the compared protocols (i.e., V-NDN, CCLF, and multicast). But, it presents high retransmission than CMAF with flooding. This results from the fact that elongating packets with additional information,

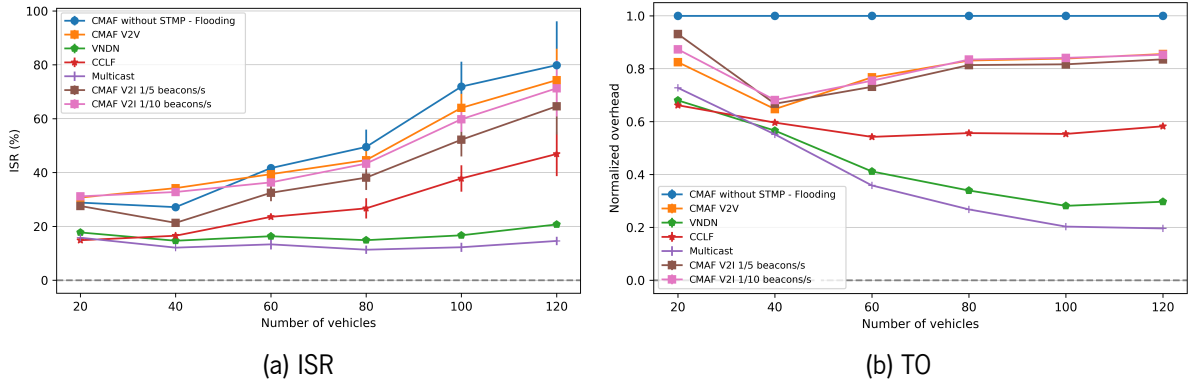


Figure 33: ISD and TO, for the Manhattan-like grid.

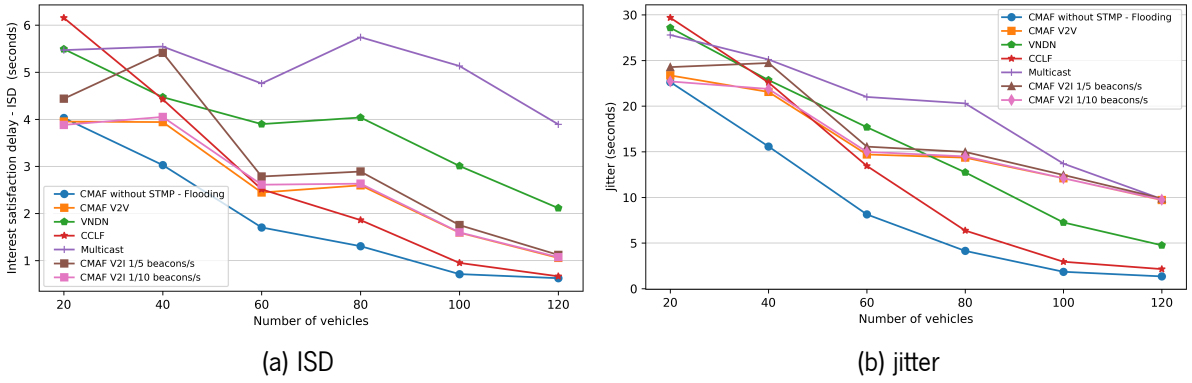


Figure 34: ISD and jitter for the selected protocols, for the Manhattan-like grid.

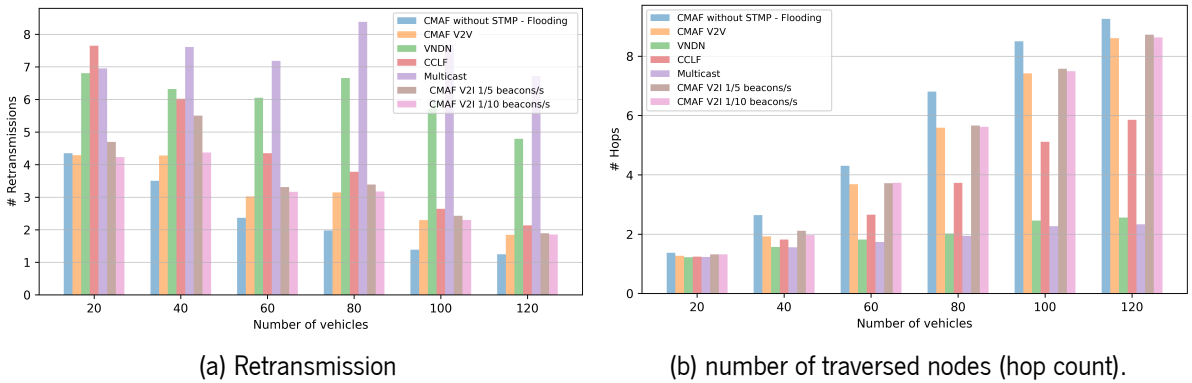


Figure 35: Average retransmissions and hop count presented by the protocols.

increases delay and the probability of incurring in transmission errors, forcing packet retransmissions.

6.3.3 Discussion at the Data Plane Level

We developed CMAF, a context- and mobility-aware forwarding strategy, aiming at reducing TO while keeping the performance high. The proposed protocol works in two modes, V2V (beaconless) and V2I

(beacon-based), and in either mode, it outperforms compared state-of-the-art solutions, at a cost of an increased ISD. To further reduce the transmission overhead, a mechanism based on neighborhood density is used to limit the number of hops Data packets traverse from their provider to the consumer. The proposed mechanism still needs refinements to further adapt it with larger differences in node density and for different road topologies, as it shows a consistent effect on the results in both chosen topologies (i.e., the city of Porto's and the Manhattan-like road networks) but with different weights for different densities. In terms of improved performance vs. TO, the proposed model (CMAF V2V) achieves the same performance (i.e., ISR) as that of the flooding scenario (CMAF—flooding), with averages of 15% and 20% TO reductions for the city of Porto's and the Manhattan-like road topologies, respectively.

As previously stated, the model is designed to work in both V2V and V2I modes. CMAF V2V presents superior performance when compared to the state-of-the-art solutions; it presents 5%–10% superior ISR than the CCLF for the same overhead, at the cost of 1 s of increased ISD. Moreover, the number of retransmissions of CMAF is also reduced for relatively the same number of hops for the compared scenarios. Compared to VNDN and Multicast, CMAF presents fewer retransmissions and 10% to 45% superior ISR, with an increased overhead of about 20% in scenarios with 60 to 120 vehicles.

In V2I mode, a controlled beacon is broadcast. When the broadcast frequency increases, the performance degrades, as expected, due to the TO. For a lower beaconing frequency (i.e., one beacon every 10 s), the model still achieves better performance when compared with the state of the art.

6.4 Simulations results for MARV

This section presents the simulation results for MARV. Specifically, it presents the simulation results of the model where the data plane explicitly resort to MARV for determining the next-hop. Herein we highlight that CMAF in data plane only forwards the received Interest if: 1) a corresponding route to content provider is present in FIB or 2) the packet is new and the first being received by the node and no corresponding FIB entry is found. Otherwise, the Interest is discarded. This behavior is defined this way in order to clearly study and understand the advantages and disadvantages of MARV apart from the ones from CMAF.

Section 6.3 presents results of CMAF taking in consideration two specific environments - the city of Porto, in Section 6.3.1 and the Manhattan-like, in Section 6.3.2. The present section only considers the city of Porto road scenario. The reasoning behind this choice is that, the Manhattan-like road network presents lower level of randomness than the city of Porto road network, which includes highways and is more susceptible of having disconnected segments, as already stated in section 5.2. Therefore, we chose the former for simulating MARV, considering that the robustness of the model, for partitioned network, would be well tested. This section presents the simulation results of a) CMAF without mobility prediction; b) CMAF with mobility prediction, however, not explicitly resorting to FIB. That is, without MARV; and c) the results of the complete model, where CMAF resorts to FIB - which is explicitly managed by MARV, to select the next-hop.

Figure 36 depicts the ISR and TO of the compared solutions. As shown, CMAF benefits from the use of MARV. When using MARV, for the same ISR, CMAF is allowed to reduce TO by about 10%. As we will see next, this reduction is gained at the cost of the increase of ISD and jitter.

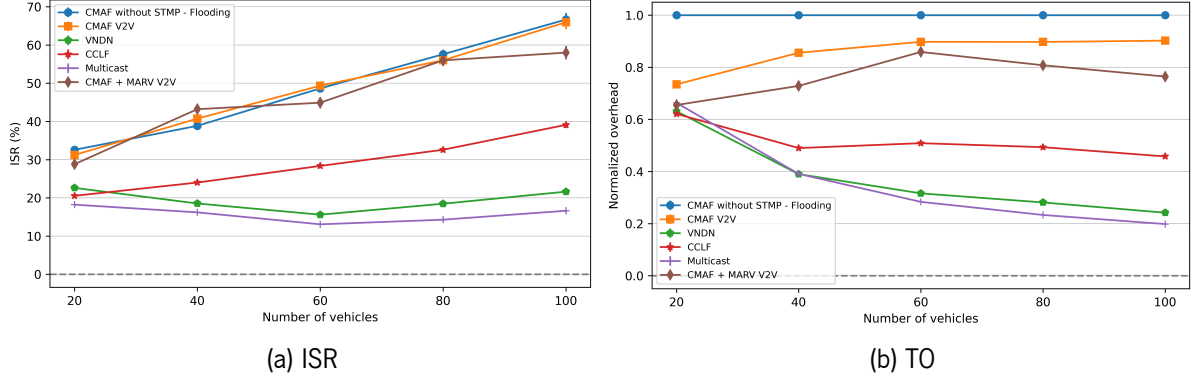


Figure 36: ISR and TO for the selected protocols without Data hop-limit, with MARV presenting the better performance, in terms of reduced TO while keeping ISR at the level of CMAF.

The referred result in terms of TO vs ISR, can be reinforced further by the results shown in Figure 37. The figure depicts the average number of transmitted Interest (Figure 37a) and Data (Figure 37b). As depicted, CMAF + MARV V2V shows a better utilization of resources as, the ISR of this configuration have same magnitude as the CMAF V2V and CMAF Flooding but, with less Interest and Data in the network.

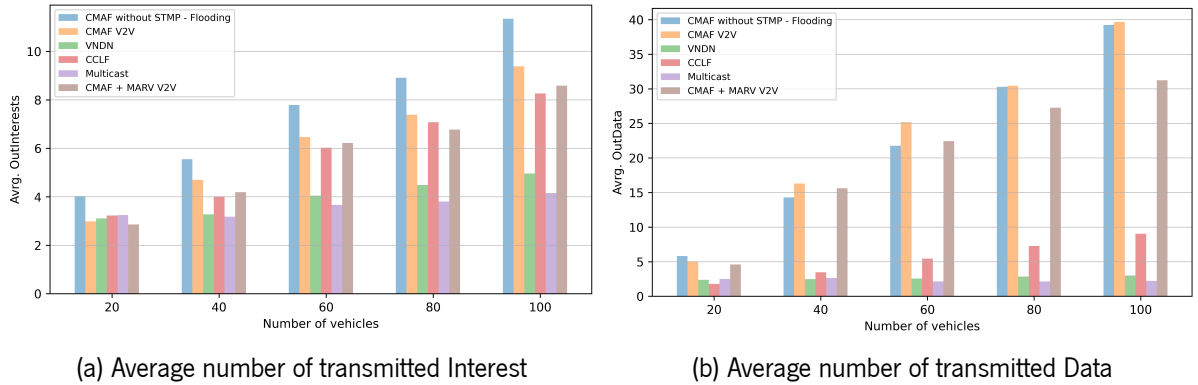


Figure 37: Average number of transmitted Interest and Data

As already referred ISD and jitter of CMAF V2V is negatively affected by the use of MARV. This is due, in part, by the increased processing overhead posed STMP on keeping FIB updated. The reactive mode mainly, where before selecting next-hop, its current geo-position is predicted to ensure its status as a still valid neighbor, introduces a not negligible delay. The results of Jitter show a similar behavior. The result is shown in Figure 38.

In terms of retransmission, Figure 39a shows that MARV affects negatively the network by allowing high retransmissions. This is due to the fact that the delay of transmitting the content results in an increased probability of incurring in transmission errors, which forces the consumer to retransmit the

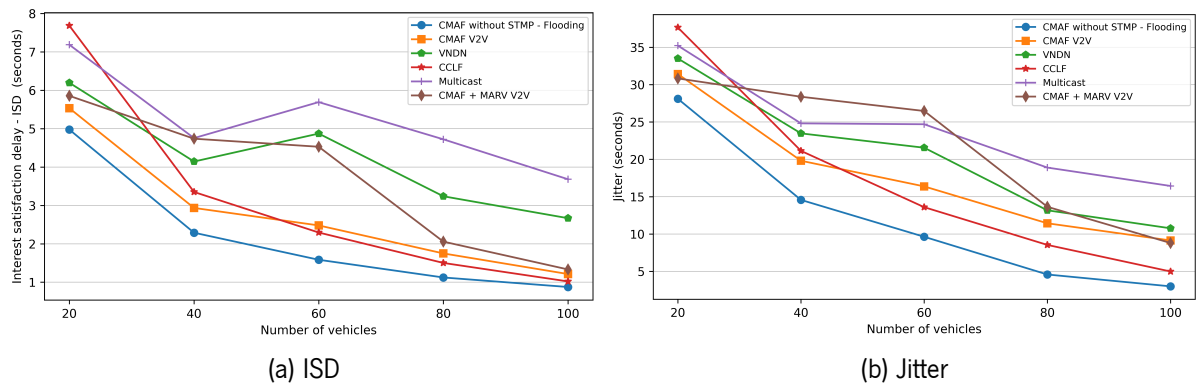


Figure 38: ISD and jitter for the selected protocols, for simulations without limiting Data hop count.

request. In terms of hop-count, as depicted in Figure 39b, CMAF presents low hop-count when relying on MARV for choosing the next-hop. This can be justified by the fact that MARV only keeps in FIB the next-hops with lower hop-count, as presented in Section 4.4.4.

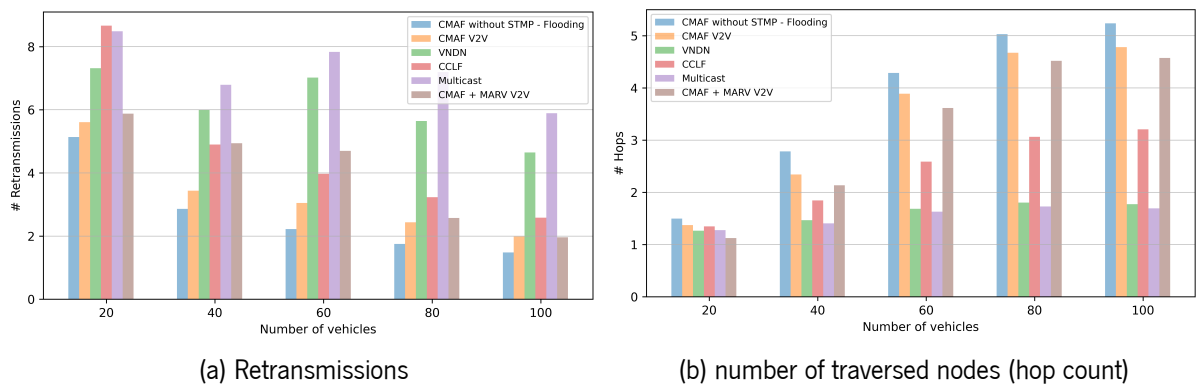


Figure 39: Average packet retransmissions and hop count presented by the protocols.

6.4.1 Discussion at the Control Plane Level

Section 6.3.3 presented the discussion of the evaluation of CMAF, considering its stateful propriety, thus not explicitly resorting to FIB for deciding which next-hop to select for forwarding. This section presented the evaluation of the same CMAF but now, resorting explicitly to FIB for choosing next-hop. In this configurations, CMAF only forwards the Interest packets if: a) there is any corresponding FIB entry from where the next-hop could be chosen. MARV keeps a number of next-hops in FIB limited by the threshold related to the network density, as presented in Section 4.4.4. For multipath purpose, CMAF forwards the Interest packet to all next-hops registered in the corresponding FIB entry; b) The received Interest is the first in all possible retransmission. This is done in order to allow the system to discover the content provider. No other Interest with the same name prefix is forwarded, until a corresponding FIB is registered or until the Interest lifetime has elapsed and the corresponding PIT entry has been removed.

By allowing CMAF to rely only on FIB we ensure that the difference in terms of network performance between the results with the system running CMAF without MARV and the ones where CMAF relies on MARV reflects the advantage or disadvantages of MARV.

Having this fact in consideration, we see, from the results presented in the previous Section 6.4, that MARV allows the reduction of TO whilst keeping the ISR the same magnitude as the results presented when CMAF does not explicitly use FIB. However, this results have some costs, which is the increase of ISD as one could expect, given the additional overhead the FIB maintenance imply. Moreover, designing MARV to only keep in FIB the next-hops with lower hop-count resulted in a reduction in terms of hop-count.

In terms of average Interest and Data in the network, MARV allows the reduction of these packets. In fact, this (their ratio) is what contribute to the TO.

It is highly desirable to evaluate the advantages of MARV further, by simulating the network with state-of-the-art forwarding strategies relying on it for populating FIB. Additionally, it should be directly compared with the state-of-the-art routing protocols. For now, we leave this investigation for future work

6.5 General Discussion

The proposed work aimed at developing a context-aware routing and forwarding model capable of achieving high performance but keeping the overhead low. For such a case, instead of broadcasting a beacon for neighbor discovery, a node leverages the overheard packet from the nodes in its reception range. Instead of dropping unsolicited or looped packets, CMAF processes all received packets to manage the NT. Although this procedure increases the computational processing overhead, it also shows its potential to reduce the overhead, and this is observed when comparing the two modes of operation and the result when the beaconing frequency increases. Moreover, the model is also designed to resort to a mobility prediction scheme for better management of the NT. With a consistent NT, only valid and capable nodes could be chosen as next-hops. This improves the performance further.

Furthermore, with CMAF V2I, we intended to investigate the impact on the performance when having further knowledge about the content in the neighbors' CS. Here, given that two beacon broadcast processes take place (i.e., one querying the content producer and the other for querying the 1-hop previous neighbor's CS catalog), the results show that the performance cannot be improved. In addition to the problem caused by the beaconing process, the received list of neighbors' CS cannot be used for a long time. Whenever a node is purged from the NT, tracking it and updating the information in the list with the current node position, for each name prefix, turns out to be infeasible. Moreover, the packet holding the neighbor's CS catalog is sent using BF. The packet carrying the BF Data ends up being an additional source of TO due to its size.

In summary, our main hypothesis was that by incorporating the mobility prediction and leveraging the characteristics of the wireless channel (broadcast by nature), we would be able to reduce the overhead

induced by the beaconing process for neighbor discovery and by the retransmissions due to the inconsistency of the NT. And, as shown in Section 6.3, the model achieved the expected result. Moreover, the results show that additional packets in the network for neighbor discovery and neighbor's CS catalog dissemination only increase the TO, degrading the overall performance.

Moreover, piggybacking the NMSI on packets results in elongating their size. If we despise other components of the nodal delay and look only on the transmission delay, we have $d_{trans} = L/R$ with L as the packet length (in bits) and R the transmission rate (in bp/s). Then, elongating the packet (i.e., increasing L) will increase the transmission delay and subsequently the probability of getting transmission errors and enforce retransmission. This fact increases ISD, as shown by the results.

6.6 Summary

This chapter presented the evaluation of the proposed model. It starts by presenting the chosen metrics for evaluation and comparison with state-of-the-art. Then, the simulation environment is presented. The simulations results of CMAF, a forwarding strategy that does not explicitly rely on FIB, are presented. Two specific scenarios are chosen for the evaluation, the road network similar to the Manhattan's and the road network of the city of Porto. CMAF is evaluated in both scenarios. Next, the simulation results of MARV are presented. The discussions are, first presented separately for CMAF and MARV and, then a general discussion is presented.

Conclusions and Future Work

This chapter aims at presenting the main conclusions of the thesis. Initially a brief summary is presented in Section 7.1. Then, Research questions, which guided the development of this work, are revisited in order to draw conclusions answering them, as presented in Section 7.2. Additionally, the main limitations and recommendations are presented in Section 7.3. Also, the future work is presented, in Section 7.4.

7.1 Summary

The present thesis proposes a context- and mobility-aware model for NDN-based VANET aimed advancing the integration of NDN-based VANET in ITS by providing an efficient communication model that closes current gaps, extracted from the state-of-the-art, in content dissemination in NDN-based VANET. The model is designed and developed in two steps: a) by means of a forwarding strategy, at the data plane, dubbed CMAF and b) a subsequent development of a routing model, at the control plane, dubbed MARV.

- a) The strategy is designed to consider the network context on its decision for choosing the next-hop, and resort to mobility prediction to proactively keep NT updated. It leverages the overheard packets to feed the mobility prediction algorithm (i.e., the STMP) and in one of its communication mode - V2I, it takes advantages of the existing RSU, which resorting to a periodic beaconing process requests the catalogues of the CS of the nodes on the forwarding path.
- b) The routing model also leverages the overheard packets to feed the mobility prediction algorithm (i.e., the STMP), which is used to proactively keep FIB updated and, reactively update routes on demand.

We highlight that, as presented in Section 5.5, STMP was initially proposed for the forwarding plane and LTMP for the routing plane. At last, both planes used STMP in all mobile nodes, and LTMP was only used by the static nodes, i.e., the RSUs.

By means of extensive simulations, the model was evaluated and the results shows its superiority in terms of reduced TO whilst keeping the ISR high, at a cost of relatively high ISD resulting from the overhead,

mainly induced by the mobility prediction processing and the increased size of packets due to the inclusion of NMSI, *origID* and *destFinalID* on Interest and Data packets, as presented in Section 5.4.1.2.

7.2 Conclusions and Answers to the Research Questions

In Chapter 1 we presented the motivation and the research questions which guided the development of the present thesis. The main objective of the thesis is to answer these research questions. The methodology adopted to attain the objective was presented in that chapter, in summary, it is as follows:

- In RQ1 we investigated *whether integrating mobility prediction could reduce network traffic and provide better utilization of the network resources*. To answer this question we proposed two mobility prediction algorithms: a) the STMP at the forwarding plane, and the b) LTMP at the routing plane.
 - a) The definition and the design decisions of STMP are presented in Section 2.4 and Section 4.5. The algorithm is reactively invoked by the forwarding strategy whenever a forwarding node should be selected, amongst the existing neighbors in the NT. It also runs proactively as described and justified in Section 4.3.2.1. As shown by the results from Section 6.3, the performance of CMAF, in terms of ISR, without the use of STMP is of the same magnitude as the CMAF with the use of STMP. However, the TO of CMAF using STMP is at least 10% below the value presented by the strategy without mobility prediction. The referred result is achieved at the cost of aggravated ISD, which was in fact expected considering the computational effort the system had to make in order to predict the mobility of nodes. We highlight that given the fact that the model is hybrid, in the sense that it predicts the mobility on demand (reactively) and proactively, only the reactive mode really contributes to the increase of ISD.
 - b) The definition and the design decisions of LTMP are presented in Section 2.4 and Section 4.6. The algorithm is reactively invoked by the routing protocol whenever a forwarding node should be selected, amongst the existing neighbors in the FIB. Similar to STMP algorithm, LTMP also runs in hybrid mode and it proactively predicts the mobility of nodes that corresponds to the IDs registered in FIB entries.

In summary, as shown by the results presented in Chapter 6, the inclusion of mobility prediction, either at the forwarding plane or at the routing plane results in a reduction of TO at the cost of increased ISD due to the mobility prediction processing overhead. Thus, responding to the posed research question, we conclude that, although at the cost of increased ISD mainly imposed by the reactive mode of mobility prediction, the inclusion of mobility prediction does indeed reduce TO, whilst maintaining ISR high, contributing this way to the better utilization of the network resources. We recall the discussion in several studies, as presented in previous chapters (see, for instance, Section 2.2.2), which concluded that routing is infeasible in networks with highly intermittent routes.

Given the current technological developments, in terms computer system with more and more processing power, and the development of new prediction models based on artificial intelligence, the results achieved by the present work are in line with several other studies that conclude that routing, and we now say specially routing based on mobility prediction, can still play a pivotal role on increasing the content dissemination performance in VANET.

- In RQ2, we investigated *at what extent the awareness of the network environment could play a role for efficient routing in VANET*. Taking in consideration one of the functional requirements for a good routing protocol (i.e., one that must be able to adapt itself to the specific VANET environments), as presented by [200], we designed the routing model with the capability of deciding on which next-hop to forward packets, taking in consideration various environment factors such as: a) the type of content, b) the communication models, and c) the network density.
 - a) The model adapts its content dissemination mode (i.e., unicast, broadcast) considering the “type of messages” to forward. It immediately broadcasts the active safety- and efficiency-related messages. It processes for forwarding, via unicast, some of efficiency-related packets and the comfort- or interactive-entertainment-related packets. Moreover, as explained in Section 4.3.3.1 it also uses the “type of messages” for deciding which content, including the unsolicited, to cache. The broadcasting of active safety- and efficiency-related messages favors the transmission of these messages without the delay that could have been introduced by the routing processing.
 - b) It bases its decision on whether the environment should be taken as urban or rural, considering the speed of the node and the presence or not of RSUs. The geo-locations of RSUs are persistently stored in separate NT - The NTRsu. Given that entries in NTRsu never get removed, nodes have an anytime means of knowing whether there is or not an RSU in its vicinity. The model takes advantage of the existing RSU to send to it the Interest whose destination is unknown (i.e., Interest to which a corresponding FIB entry does not exist). If the forwarding node’s speed is below a defined threshold of $scalar_speed = 16.67m/s$, and there is any RSU in its vicinity, then, it considers the environment as an urban scenario and consequently the V2V mode is selected for data routing. That is, for urban scenario, the communication model to adopt is V2V, otherwise, the V2I mode is selected. We highlight that the selection of V2I mode depends on the existence of RSU in the network. As the results show, the model presents better performance in urban scenarios running in V2V mode. Being aware of the network environment, enables the model to select the best model to efficiently route packets to the destination.
 - c) The model is also aware of the network density and, in that conformity, it selects the best and most adequate properties (i.e., maximum number of hops) for better routing. It accesses this

factor by looking at its updated NT, as described in Section 4.3.3.2. For a crowded environment, the number of hops a specific packet is disseminated should be adequately defined, otherwise the cost of keeping the network free of loops could be high. Section 6.3.1.3 shows the effect of choosing adequate number of hops, considering the network density. The results further illustrates that limiting the number of hops for different packets (i.e., Interest or Data) affects differently the network performance. The fact is justified by the size of the packet, as explained in the aforementioned section.

In summary, and answering the posed research question, we conclude that the awareness of the network environment is crucial for better routing decision, as the different properties of the environment affects differently the network performance.

- In RQ3, we aimed at answering *to what extent selectively sharing cached content among vehicles for further helping the routing decision would reduce the need for broadcasting and hence reduce the network traffic, when no FIB entry matches the requested content* - In rural scenarios, running in V2I mode the proposed solution relies on RSU for sending periodic beacons in order to gather the catalogues of neighbors's CS contents. Thus, differently from the V2V mode, although less frequent, in V2I mode the model sends periodic beacons for selected neighbors. The procedure is detailed in Section 4.3.2.2. Results showed that, although sending beacon only to a few selecting neighbors, the TO induced by these additional beacons is enough to negatively affect the network performance. In fact, as shown, the higher is the beaconing frequency, the worse is the network performance.

In summary, we conclude that the addition of new beacons increases the network load, hiding the potential benefit of having knowledge of the CS content of neighbors. We now speculate that, given a means of gathering neighbors' CS catalogs without increasing the network load further, through additional beacon packets, it could still be beneficial for a better routing decision.

7.3 Main Limitations and Recommendations

This section presents the main limitations encountered during the execution of the present work.

- Generally, in order to correctly assess the contribution of the solution one proposes, it is necessary to compare it against the state-of-the-art. In order to make the comparison, the chosen state-of-the art solution for comparison should be deployed in the system and be simulated in the same environment as the proposed solution. Unfortunately, it is not easy to access the source code of the existing solutions, and getting it from the Authors turns out to be not as easy as it should be. For forwarding strategies, this task is easy given the existence of a clear methodology for designing a forwarding strategy. If we consider the lack of detailed descriptions of the implementations,

which is a general issue on the surveyed state-of-the-art, this limitation gets more aggravated. The recommendation here is to stimulate researches to share their artefacts with the research communities, whenever it is possible.

- Another limitation is attached to computational resources for simulations. As explained in Section 5.5, the system used to run simulations was not always up to the task, mainly when simulating MARV. It would have been much easier and less stressing to simulate and extract the produced results in a system with better specifications. The recommendations here is to acquire licenses or other means for using systems such as the case of Deucalion, the super computer installed at University of Minho.

7.4 Future Work

The work presented in this thesis has shown the impact of using mobility prediction either in data and control planes, for boosting content delivery in VANET by keeping the TO low. Several aspects have been learned during the execution of the proposed work. Some other aspects have been left open for future investigations. This section presents some of them.

- After deploying MARV, CMAF only resorts to it for forwarding packets. That is, resorts to FIB for choosing the next-hop. More simulations are needed to compare the results where CMAF also use its stateful property for deciding on the next-hop when FIB has no corresponding entry for the received Interest.
- MARV and CMAF are developed in the same class. This should be done in separated classes, in order to be able to evaluate MARV with different forwarding strategies (e.g., the best-route strategy, already installed in ndnSIM), using the methodology similar to the one used in [261]. It could be installed as an application, which is installed in each node in the network. The application should not issue any Interest, instead, it should run an endless method that works with the developed STMP, predicting the mobility of nodes and updating FIB.
- The beneficial advantage of MARV, in terms of reducing TO has been presented in Section 6.4, compared to the results presented in Section 6.3. Further simulations are desirable, where the model should be compared to the state-of-the-art (i.e., routing solutions). This is intimately related to one of the presented limitations, in Section 7.3.
- Section 2.1.4 briefly presented the security, privacy and the trust aspects of VANET. Those aspects are out of scope of this thesis. They are, however, of a paramount importance for VANET. First, we propose mobility prediction schemes aiming at tracking the mobility of nodes. Moreover, we propose the inclusion of an identity of the packet sending node, which is a form of exposing the vehicle identity. These aspects fall into the common VANET privacy and security vulnerabilities, as

presented in the referred section and in Section 3.9.3. Specific issues on privacy and security in VANET can easily be a subjects for an entire thesis. However, it is still possible to tackle some of them at the level of an specific objective in works like the present. As a proposal for future work, we can include the advising of new mechanisms to address the new security issues that arise by including the node ID in the NDN packets, which could be, for instance, coding the ID, and other additional information capable of exposing the sender, into CN.

- To better access the performance of the proposed model, simulations could be extended to include factors such as: a) producer popularity, b) content popularity, c) leveraging fog computing or vehicle platooning for offload processing, to help running the mobility prediction algorithm. In this regard, even machine learning algorithms, which are more accurate for predicting node mobility but, more demanding in terms of processing power, could be tested.

Bibliography

- [1] N. Abani, T. Braun, and M. Gerla. “Proactive Caching with Mobility Prediction under Uncertainty in Information-Centric Networks”. In: *Proceedings of the 4th ACM Conference on Information-Centric Networking*. ICN '17. Berlin, Germany: Association for Computing Machinery, 2017, pp. 88–97. isbn: 9781450351225. doi: 10.1145/3125719.3125728 (cit. on pp. 43, 52, 53, 64, 65, 67).
- [2] I. Abdel-Halim and H. Fahmy. “Prediction-based Protocols for Vehicular Ad Hoc Networks: Survey and Taxonomy”. In: *Computer Networks* 130 (2017-10). doi: 10.1016/j.comnet.2017.10.009 (cit. on pp. 9, 26, 101).
- [3] M. Abdullah et al. “Interest flooding attack mitigation in a vehicular named data network”. In: *IET Intelligent Transport Systems* 15.4 (2021), pp. 525–537. doi: <https://doi.org/10.1049/itr2.12042> (cit. on pp. 45, 57, 58).
- [4] ACM. *ACM Digital Library*. 2020-03. url: www.dl.acm.org (cit. on p. 41).
- [5] A. Afanasyev et al. “Interest flooding attack and countermeasures in Named Data Networking”. In: *2013 IFIP Networking Conference*. 2013, pp. 1–9 (cit. on p. 57).
- [6] A. Afanasyev et al. *NFD Developer’s Guide*. Tech. rep. 2018-02. doi: 10.13140/RG.2.2.28896.38408 (cit. on pp. 21, 106, 107, 114, 115, 117, 121, 124).
- [7] A. Agarwal and S. R. Das. “Dead reckoning in mobile ad hoc networks”. In: *2003 IEEE Wireless Communications and Networking, 2003. WCNC 2003*. Vol. 3. 2003, 1838–1843 vol.3. doi: 10.1109/WCNC.2003.1200666 (cit. on p. 27).
- [8] K. Ahed et al. “Forwarding strategies in vehicular named data networks: A survey”. In: *Journal of King Saud University - Computer and Information Sciences* (2020). issn: 1319-1578. doi: 10.1016/j.jksuci.2020.06.014 (cit. on pp. 38, 39, 73).
- [9] B. Ahlgren et al. “A survey of information-centric networking”. In: *IEEE Communications Magazine* 50.7 (2012), pp. 26–36. doi: 10.1109/MCOM.2012.6231276 (cit. on pp. 2, 19, 20).

- [10] S. H. Ahmed et al. "CONET: Controlled data packets propagation in vehicular Named Data Networks". In: *2016 13th IEEE Annual Consumer Communications Networking Conference (CCNC)*. 2016, pp. 620–625. doi: 10.1109/CCNC.2016.7444850 (cit. on pp. 38, 79).
- [11] S. H. Ahmed, S. H. Bouk, and D. Kim. "RUFs: RobUst Forwarder Selection in Vehicular Content-Centric Networks". In: *IEEE Communications Letters* 19.9 (2015), pp. 1616–1619. doi: 10.1109/LCOMM.2015.2451647 (cit. on pp. 73, 75, 77, 80).
- [12] S. H. Ahmed et al. "DIFS: Distributed Interest Forwarder Selection in Vehicular Named Data Networks". In: *IEEE Transactions on Intelligent Transportation Systems* 19.9 (2018), pp. 3076–3080. doi: 10.1109/TITS.2017.2768329 (cit. on pp. 77, 80).
- [13] H. Al-Omaisi et al. "A survey of data dissemination schemes in vehicular named data networking". In: *Vehicular Communications* 30 (2021), p. 100353. issn: 2214-2096. doi: 10.1016/j.vehcom.2021.100353 (cit. on p. 83).
- [14] H. Al-Omaisi et al. "GeolSA: A New Road-Topology-Assisted Geo-Based Content Discovery Scheme for Vehicular Named Data Networking". In: *Veh. Commun.* 40.C (2023-04). issn: 2214-2096. doi: 10.1016/j.vehcom.2023.100573 (cit. on pp. 76, 77, 80).
- [15] H. Al-Omaisi et al. "IDTracS: an Interest-Data-flow tracking-based forwarding scheme for vehicular named data networks". In: *The Journal of Supercomputing* 79 (2023-04). doi: 10.1007/s11227-023-05268-2 (cit. on pp. 76, 77, 79, 80).
- [16] S. Al-Sultan et al. "A comprehensive survey on vehicular Ad Hoc network". In: *Journal of Network and Computer Applications* 37 (2014), pp. 380 –392. issn: 1084-8045. doi: 10.1016/j.jnca.2013.02.036 (cit. on pp. 1, 15).
- [17] B. Aldahlan and Z. Fei. "Progressive Segment Routing for Vehicular Named Data Networks". In: *Advances in Information and Communication*. Ed. by K. Arai. Cham: Springer International Publishing, 2021, pp. 425–437. isbn: 978-3-030-73100-7 (cit. on pp. 45, 47, 49, 63).
- [18] N. Aljeri and A. Boukerche. "Performance evaluation of movement prediction techniques for vehicular networks". In: *2017 IEEE International Conference on Communications (ICC)*. 2017, pp. 1–6. doi: 10.1109/ICC.2017.7996756 (cit. on p. 28).
- [19] L. Almeida et al. "DCE21- Symposium on Electrical and Computer Engineering: Book of Abstracts". In: *Book of Abstracts of the 4th Symposium on Electrical and Computer Engineering*. Ed. by U. do Porto. Universidade do Porto, Faculdade de Engenharia. Departamento de Engenharia Eletrotécnica e de Computadores, Rua Dr. Roberto Frias. 4200-465 Porto, Portugal: FEUP Edições, 2021-06, pp. 56. doi: 10.24840/978-972-752-276-7 (cit. on pp. 9, 10).
- [20] M. Amadeo, C. Campolo, and A. Molinaro. "CRoWN: Content-Centric Networking in Vehicular Ad Hoc Networks". In: *IEEE Communications Letters* 16.9 (2012), pp. 1380–1383. doi: 10.1109/LCOMM.2012.072012.120282 (cit. on pp. 38, 83).

-
- [21] M. Amadeo, C. Campolo, and A. Molinaro. "Information-centric networking for connected vehicles: a survey and future perspectives". In: *IEEE Communications Magazine* 54.2 (2016), pp. 98–104. doi: 10.1109/MCOM.2016.7402268 (cit. on pp. 38, 39, 73, 83, 96).
 - [22] M. Amadeo et al. "Diversity-improved caching of popular transient contents in Vehicular Named Data Networking". In: *Computer Networks* 184 (2021). doi: 10.1016/j.comnet.2020.107625 (cit. on pp. 45, 52, 55, 69, 95).
 - [23] M. Amadeo, C. Campolo, and A. Molinaro. "Forwarding strategies in named data wireless ad hoc networks: Design and evaluation". In: *Journal of Network and Computer Applications* 50 (2015), pp. 148–158. issn: 1084-8045. doi: 10.1016/j.jnca.2014.06.007 (cit. on pp. 25, 73).
 - [24] M. Amadeo et al. "Caching Transient Contents in Vehicular Named Data Networking: A Performance Analysis". In: *Sensors* 20 (2020-04), p. 1985. doi: 10.3390/s20071985 (cit. on pp. 45, 52, 55, 69, 95).
 - [25] M. Amadeo et al. "Content-centric wireless networking: A survey". In: *Comput. Networks* 72 (2014), pp. 1–13. doi: 10.1016/j.comnet.2014.07.003 (cit. on pp. 38, 39, 90).
 - [26] C. Anastasiades, J. Weber, and T. Braun. "Dynamic Unicast: Information-centric multi-hop routing for mobile ad-hoc networks". In: *Computer Networks* 107 (2016), pp. 208–219. doi: 10.1016/j.comnet.2016.03.009 (cit. on pp. 43, 47, 99).
 - [27] S. P. Ardakani et al. "CNN: A Cluster-based Named Data Routing for Vehicular Networks". In: *IEEE Access* (2021), pp. 1–1. doi: 10.1109/ACCESS.2021.3131198 (cit. on pp. 45, 47, 49, 63).
 - [28] W. T. Ariefianto and N. R. Syambas. "Routing in NDN network: A survey and future perspectives". In: *2017 11th International Conference on Telecommunication Systems Services and Applications (TSSA)*. 2017, pp. 1–6. doi: 10.1109/TSSA.2017.8272942 (cit. on p. 38).
 - [29] G. Arnould, D. Khadraoui, and Z. Habbas. "A Self-Organizing Content Centric Network Model for Hybrid Vehicular Ad-Hoc Networks". In: *Proceedings of the First ACM International Symposium on Design and Analysis of Intelligent Vehicular Networks and Applications*. DIVANet '11. Miami, Florida, USA: Association for Computing Machinery, 2011, pp. 15–22. isbn: 9781450309042. doi: 10.1145/2069000.2069004 (cit. on pp. 38, 43, 47, 49, 61, 83).
 - [30] M. M. Artimy, W. Robertson, and W. J. Phillips. "Vehicular Ad Hoc Networks: An Emerging Technology Toward Safe and Efficient Transportation". In: *Algorithms and Protocols for Wireless and Mobile Ad Hoc Networks*. John Wiley and Sons, Ltd, 2008. Chap. 14, pp. 405–432. isbn: 9780470396384. doi: 10.1002/9780470396384.ch14 (cit. on p. 108).
 - [31] M. Awais and M. A. Shah. "Information-centric networking: A review on futuristic networks". In: *2017 23rd International Conference on Automation and Computing (ICAC)*. 2017, pp. 1–5. doi: 10.23919/ICoNAC.2017.8082033 (cit. on p. 20).

- [32] L. N. Balico et al. "Localization Prediction in Vehicular Ad Hoc Networks". In: *IEEE Communications Surveys Tutorials* 20.4 (2018), pp. 2784–2803. doi: 10.1109/COMST.2018.2841901 (cit. on p. 28).
- [33] L. N. Balico et al. "On the performance of localization prediction methods for vehicular Ad Hoc Networks". In: *2015 IEEE Symposium on Computers and Communication (ISCC)*. 2015, pp. 359–364. doi: 10.1109/ISCC.2015.7405541 (cit. on p. 27).
- [34] Y. Bang, J. Kim, and K. Yu. "An Improved Map-Matching Technique Based on the Fréchet Distance Approach for Pedestrian Navigation Services". In: *Sensors* 16 (2016-10), p. 1768. doi: 10.3390/s16101768 (cit. on p. 103).
- [35] M. F. Bari et al. "A survey of naming and routing in information-centric networks". In: *IEEE Communications Magazine* 50.12 (2012), pp. 44–53. doi: 10.1109/MCOM.2012.6384450 (cit. on p. 38).
- [36] P. Basaras and D. Katsaros. "Detecting Influential Nodes in Complex Networks with Range Probabilistic Control Centrality". In: vol. 456. 2015-09. doi: 10.1007/978-3-319-10407-2_32 (cit. on p. 56).
- [37] L. E. Baum et al. "A maximization technique occurring in the statistical analysis of probabilistic functions of Markov chains". In: *Annals of Mathematical Statistics* 41 (1970), pp. 164–171 (cit. on p. 34).
- [38] L. E. Baum and T. Petrie. "Statistical inference for probabilistic functions of finite state Markov chains". In: *The Annals of Mathematical Statistics* (1966), pp. 1554–1563 (cit. on p. 34).
- [39] A. Benmoussa et al. "MSIDN: Mitigation of Sophisticated Interest Flooding-Based DDoS Attacks in Named Data Networking". In: *Future Gener. Comput. Syst.* 107.C (2020-06), pp. 293–306. issn: 0167-739X. doi: 10.1016/j.future.2020.01.043 (cit. on p. 57).
- [40] S. S. Beri and N. Dutta. "Evolutionary game based interest forwarding in information centric vehicular networks". In: *Vehicular Communications* 47 (2024), p. 100779. issn: 2214-2096. doi: 10.1016/j.vehcom.2024.100779 (cit. on pp. 73, 80).
- [41] C. Bernardini, M. R. Asghar, and B. Crispo. "Security and privacy in vehicular communications: Challenges and opportunities". In: *Vehicular Communications* 10 (2017), pp. 13–28. issn: 2214-2096. doi: 10.1016/j.vehcom.2017.10.002 (cit. on p. 17).
- [42] C. Bian et al. "Boosting named data networking for data dissemination in urban VANET scenarios". In: *Vehicular Communications* 2.4 (2015), pp. 195 –207. issn: 2214-2096. doi: 10.1016/j.vehcom.2015.08.001 (cit. on pp. 43, 52, 55, 66).
- [43] J. Biolchini et al. "Systematic Review in Software Engineering". In: (2005-01) (cit. on p. 40).

-
- [44] S. Bitam and A. Mellouk. "Vehicular Ad Hoc Networks". In: *Bio-Inspired Routing Protocols for Vehicular Ad Hoc Networks*. John Wiley and Sons, Inc., 2014-09, pp. 1–27. doi: 10.1002/9781119004967.ch1 (cit. on p. 98).
 - [45] B. Bloom. "Space/Time Trade-offs in Hash Coding with Allowable Errors". In: *Commun. ACM* 13.7 (1970-07), pp. 422–426. issn: 0001-0782. doi: 10.1145/362686.362692 (cit. on pp. 10, 48).
 - [46] S. H. Bouk, S. H. Ahmed, and D. Kim. "Vehicular Content Centric Network (VCCN): A Survey and Research Challenges". In: *Proceedings of the 30th Annual ACM Symposium on Applied Computing*. SAC '15. Salamanca, Spain: Association for Computing Machinery, 2015, pp. 695–700. isbn: 9781450331968. doi: 10.1145/2695664.2695844 (cit. on pp. 25, 38, 39, 83).
 - [47] A. Boukerche, R. W. L. Coutinho, and X. Yu. "LISIC: A Link Stability-Based Protocol for Vehicular Information-Centric Networks". In: *2017 IEEE 14th International Conference on Mobile Ad Hoc and Sensor Systems (MASS)*. 2017, pp. 233–240. doi: 10.1109/MASS.2017.34 (cit. on pp. 62, 77, 78, 80).
 - [48] A. Boukerche and R. W. Coutinho. "LoICen: A novel location-based and information-centric architecture for content distribution in vehicular networks". In: *Ad Hoc Networks* 93 (2019), p. 101899. issn: 1570-8705. doi: 10.1016/j.adhoc.2019.101899 (cit. on pp. 77, 78, 80).
 - [49] S. Boussoufa-Lahlah, F. Semchedine, and L. Bouallouche-Medjkoune. "Geographic routing protocols for Vehicular Ad hoc NETWORKS (VANETs): A survey". In: *Vehicular Communications* 11 (2018), pp. 20–31. issn: 2214-2096. doi: 10.1016/j.vehcom.2018.01.006 (cit. on p. 98).
 - [50] T. Braun, D. Careglio, and I. Matta. "Vehicular Networking in the Recursive InterNetwork Architecture". In: *2018 IEEE 87th Vehicular Technology Conference (VTC Spring)*. 2018, pp. 1–5. doi: 10.1109/VTCSpring.2018.8417755 (cit. on pp. 2, 5, 89, 99).
 - [51] L. Breslau et al. "Web caching and Zipf-like distributions: evidence and implications". In: *IEEE INFOCOM '99. Conference on Computer Communications. Proceedings. Eighteenth Annual Joint Conference of the IEEE Computer and Communications Societies. The Future is Now (Cat. No.99CH36320)*. Vol. 1. 1999-03, 126–134 vol.1. doi: 10.1109/INFCOM.1999.749260 (cit. on p. 107).
 - [52] L. Campanile et al. "Computer Network Simulation with ns-3: A Systematic Literature Review". In: *Electronics* 9.2 (2020). issn: 2079-9292. doi: 10.3390/electronics9020272 (cit. on pp. 106, 122).
 - [53] T. Cao et al. "Family-Aware Pricing Strategy for Accelerating Video Dissemination over Information-Centric Vehicular Networks". In: *2018 IEEE International Conference on Communications (ICC)*. 2018, pp. 1–7. doi: 10.1109/ICC.2018.8422411 (cit. on pp. 44, 47, 62).

- [54] A. A. V. Castro, G. Di Marzo Serugendo, and D. Konstantas. "Hovering Information - Self-Organising Information that Finds Its Own Storage". In: *2008 IEEE International Conference on Sensor Networks, Ubiquitous, and Trustworthy Computing (sutc 2008)*. 2008, pp. 193–200. doi: 10.1109/SUTC.2008.62 (cit. on p. 72).
- [55] M. H. center. *Understanding Kalman Filters*. url: https://www.mathworks.com/support/search.html?fq=asset__type__name:video%20category:control/state-estimation&page=1&s_tid=CRUX_topnav (cit. on pp. 29, 31).
- [56] S. Chakrabarti et al. *Data Mining Curriculum: A Proposal (Version 1.0)*. Technical Report. Intensive Working Group of ACM SIGKDD Curriculum Committee, 2006 (cit. on p. 28).
- [57] T. Chatterjee, S. Ruj, and S. Das Bit. "Security Issues in Named Data Networks". In: *Computer* 51 (2018-01), pp. 66–75. doi: 10.1109/MC.2018.1151010 (cit. on pp. 38, 57).
- [58] C. Chen et al. "Caching in Vehicular Named Data Networking: Architecture, Schemes and Future Directions". In: *IEEE Communications Surveys Tutorials* 22.4 (2020), pp. 2378–2407. doi: 10.1109/COMST.2020.3005361 (cit. on p. 39).
- [59] C. Chen et al. "An Intelligent Caching Strategy Considering Time-Space Characteristics in Vehicular Named Data Networks". In: *IEEE Transactions on Intelligent Transportation Systems* (2021), pp. 1–13. doi: 10.1109/TITS.2021.3128012 (cit. on pp. 2, 45, 52, 55, 69).
- [60] M. Chen et al. "VENDNET: VEhicular Named Data NETwork". In: *Vehicular Communications* 1.4 (2014), pp. 208–213. issn: 2214-2096. doi: 10.1016/j.vehcom.2014.09.002 (cit. on p. 38).
- [61] Q. Chen et al. "Transport Control Strategies in Named Data Networking: A Survey". In: *IEEE Communications Surveys Tutorials* 18.3 (2016), pp. 2052–2083. doi: 10.1109/COMST.2016.2528164 (cit. on p. 38).
- [62] C. Cheng, R. Jain, and E. van den Berg. "Location Prediction Algorithms for Mobile Wireless Systems". In: *Wireless Internet Handbook: Technologies, Standards, and Application*. USA: CRC Press, Inc., 2003, pp. 245–263. isbn: 0849315026 (cit. on p. 27).
- [63] M. Chowdhury, A. Gawande, and L. Wang. "Anonymous Authentication and Pseudonym-Renewal for VANET in NDN". In: *Proceedings of the 4th ACM Conference on Information-Centric Networking*. ICN '17. Berlin, Germany: Association for Computing Machinery, 2017, pp. 222–223. isbn: 9781450351225. doi: 10.1145/3125719.3132111 (cit. on pp. 44, 57).
- [64] M. Chowdhury, A. Gawande, and L. Wang. "Secure Information Sharing among Autonomous Vehicles in NDN". In: *2017 IEEE/ACM Second International Conference on Internet-of-Things Design and Implementation (IoTDI)*. 2017, pp. 15–26 (cit. on pp. 44, 57).

-
- [65] M. Chowdhury, J. A. Khan, and L. Wang. “Leveraging Content Connectivity and Location Awareness for Adaptive Forwarding in NDN-Based Mobile Ad Hoc Networks”. In: *Proceedings of the 7th ACM Conference on Information-Centric Networking*. ICN '20. Virtual Event, Canada: Association for Computing Machinery, 2020, pp. 59–69. isbn: 9781450380409. doi: 10.1145/3405656.3418713 (cit. on pp. 76, 80, 121).
 - [66] S. Corson and J. Macker. *Mobile Ad hoc Networking (MANET): Routing Protocol Performance Issues and Evaluation Considerations*. RFC 2501. RFC Editor, 1999-01 (cit. on p. 1).
 - [67] R. Coutinho, A. Boukerche, and X. Yu. “A Novel Location-Based Content Distribution Protocol for Vehicular Named-Data Networks”. In: *2018 IEEE Symposium on Computers and Communications (ISCC)*. 2018, pp. 01007–01012 (cit. on pp. 44, 47, 49, 62).
 - [68] F. Cunha et al. “Data communication in VANETs: Protocols, applications and challenges”. In: *Ad Hoc Networks* 44 (2016), pp. 90–103. issn: 1570-8705. doi: 10.1016/j.adhoc.2016.02.017 (cit. on pp. 15, 16).
 - [69] I. Cunha et al. “VNDN-Fuzzy - A strategy to mitigate the forwarding interests broadcast storm problem in VNDN networks”. In: 2023-01, pp. 263–270. doi: 10.1109/ICOIN56518.2023.10049030 (cit. on pp. 2, 76–78, 80).
 - [70] C. Dannewitz et al. “Network of Information (NetInf) - An Information-centric Networking Architecture”. In: *Comput. Commun.* 36.7 (2013-04), pp. 721–735. issn: 0140-3664. doi: 10.1016/j.comcom.2013.01.009 (cit. on p. 19).
 - [71] J. Day, I. Matta, and K. Mattar. “Networking is IPC: A Guiding Principle to a Better Internet”. In: *Proceedings of the 2008 ACM CoNEXT Conference*. CoNEXT '08. Madrid, Spain: Association for Computing Machinery, 2008. isbn: 9781605582108. doi: 10.1145/1544012.1544079 (cit. on pp. 19, 20).
 - [72] C. De Castro, C. Raffaelli, and O. Andrisano. “A dynamic hierarchical VANET architecture for Named Data Networking applications”. In: *2015 IEEE International Conference on Communications (ICC)*. 2015-06, pp. 3659–3665. doi: 10.1109/ICC.2015.7248893 (cit. on pp. 38, 83).
 - [73] G. Deng et al. “Distributed Probabilistic Caching strategy in VANETs through Named Data Networking”. In: *2016 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*. 2016-04, pp. 314–319. doi: 10.1109/INFCOMW.2016.7562093 (cit. on pp. 43, 52, 55, 66).
 - [74] J. Deng, S. Wang, and W. Wu. “Vehicular Named Data Networking Based on Efficient Incremental Route Update”. In: *2018 IEEE SmartWorld, Ubiquitous Intelligence Computing, Advanced Trusted Computing, Scalable Computing Communications, Cloud Big Data Computing, Internet of People and Smart City Innovation (SmartWorld/SCALCOM/UIC/ATC/CBDCom/IOP/SCI)*. 2018-10, pp. 1081–1086. doi: 10.1109/SmartWorld.2018.00188 (cit. on pp. 44, 47, 49, 62).

- [75] M. K. Denko. "Mobility Prediction Schemes in Wireless Ad Hoc Networks". In: *Advanced Wired and Wireless Networks*. Ed. by T. A. Wysocki, A. Dadej, and B. J. Wysocki. Boston, MA: Springer US, 2005, pp. 171–186. isbn: 978-0-387-22792-4. doi: 10.1007/0-387-22792-X_9 (cit. on p. 27).
- [76] L. Dong and R. Li. "Topic Based Clustering of Vehicles for Information Retrieval and Sharing". In: *2018 Tenth International Conference on Ubiquitous and Future Networks (ICUFN)*. 2018, pp. 256–261. doi: 10.1109/ICUFN.2018.8436681 (cit. on pp. 44, 47, 48, 62, 97).
- [77] W. Drira and F. Filali. "A Pub/Sub extension to NDN for efficient data collection and dissemination in V2X networks". In: *Proceeding of IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks 2014*. 2014, pp. 1–7. doi: 10.1109/WoWMoM.2014.6918991 (cit. on pp. 38, 83).
- [78] A. Dua, N. Kumar, and S. Bawa. "A systematic review on routing protocols for Vehicular Ad Hoc Networks". In: *Vehicular Communications* 1.1 (2014), pp. 33 –52. issn: 2214-2096. doi: <https://doi.org/10.1016/j.vehcom.2014.01.001> (cit. on pp. 46, 59, 98).
- [79] A. Dua et al. "Bloom Filter Based Efficient Caching Scheme for Content Distribution in Vehicular Networks". In: *2019 IEEE International Conference on Communications Workshops (ICC Workshops)*. 2019-05, pp. 1–6. doi: 10.1109/ICCW.2019.8756669 (cit. on pp. 44, 52).
- [80] M. Duan et al. "Neighbor Cache Explore Routing Protocol for VANET based on Trajectory Prediction". In: *2018 IEEE 3rd Advanced Information Technology, Electronic and Automation Control Conference (IAEAC)*. 2018, pp. 771–776 (cit. on pp. 44, 47, 48, 62, 95, 97).
- [81] J. Duarte, T. Braun, and L. Villas. "MobiVNDN: A distributed framework to support mobility in vehicular named-data networking". In: *Ad Hoc Networks* 82 (2019), pp. 77–90. doi: 10.1016/j.adhoc.2018.08.008 (cit. on pp. 44, 47, 50, 62).
- [82] J. a. Duarte et al. "A Multi-Pronged Approach to Adaptive and Context-Aware Content Dissemination in VANETs". In: *Mobile Networks and Applications* (2018-01). doi: 10.1007/s11036-017-0816-y (cit. on pp. 44, 52, 56, 67).
- [83] ETSI. *ETSI ITS Specification TR 102 638 V1.1.1. Vehicular Communications; Basic Set of Applications; Definitions*. Tech Report. European Telecommunications Standards Institute (ETSI), 2009 (cit. on pp. 15, 16).
- [84] S. Fang and H. Mao. "A Connectivity-Aware Caching Algorithm for Vehicular Content Centric Networks with Cache-Enabled Vehicles". In: 2018-08, pp. 232–236. doi: 10.1109/ICCChinaW.2018.8674477 (cit. on pp. 44, 52, 54, 67).
- [85] S. Fayyaz et al. "Information-Centric Mobile Networks: A Survey, Discussion, and Future Research Directions". In: *IEEE Access* 11 (2023), pp. 40328–40372. doi: 10.1109/ACCESS.2023.3268775 (cit. on p. 19).

-
- [86] H. Feng et al. "Location Prediction of Vehicles in VANETs Using A Kalman Filter". In: *Wireless Personal Communications* 80 (2014-08), pp. 543–559. doi: 10.1007/s11277-014-2025-3 (cit. on p. 28).
 - [87] N. Fotiou et al. "Developing information networking further: from PSIRP to PURSUIT". In: vol. 66. 2010-10. isbn: 978-3-642-30375-3. doi: 10.1007/978-3-642-30376-0_1 (cit. on p. 19).
 - [88] R. Frederick et al. *RTP: A Transport Protocol for Real-Time Applications*. RFC 1889. 1996-01. doi: 10.17487/RFC1889 (cit. on p. 121).
 - [89] S. Gambs, M.-O. Killijian, and M. N. del Prado Cortez. "Next Place Prediction Using Mobility Markov Chains". In: *Proceedings of the First Workshop on Measurement, Privacy, and Mobility*. MPM 12. Bern, Switzerland: Association for Computing Machinery, 2012. isbn: 9781450311632. doi: 10.1145/2181196.2181199 (cit. on p. 28).
 - [90] P. Gasti et al. "DoS and DDoS in Named Data Networking". In: *2013 22nd International Conference on Computer Communication and Networks (ICCCN)*. 2013, pp. 1–7. doi: 10.1109/ICCCN.2013.6614127 (cit. on p. 57).
 - [91] C. Ghasemi et al. "MUCA: New Routing for Named Data Networking". In: 2018-05, pp. 289–297. doi: 10.23919/IFIPNetworking.2018.8697026 (cit. on p. 24).
 - [92] S. Gillani et al. "A Survey on Security in Vehicular Ad Hoc Networks". In: vol. 7865. 2013-05. isbn: 978-3-642-37973-4. doi: 10.1007/978-3-642-37974-1_5 (cit. on p. 17).
 - [93] *Google Scholar*. 2020-03. url: <https://scholar.google.com/> (cit. on p. 41).
 - [94] E. Grasa et al. "Recursive InterNetwork Architecture, Investigating RINA as an Alternative to TCP/IP (IRATI)". In: 2016-01 (cit. on pp. 19, 20).
 - [95] G. Grassi et al. "Navigo: Interest forwarding by geolocations in vehicular Named Data Networking". In: *2015 IEEE 16th International Symposium on "A World of Wireless, Mobile and Multimedia Networks" (WoWMoM)*. Los Alamitos, CA, USA: IEEE Computer Society, 2015-06, pp. 1–10. doi: 10.1109/WoWMoM.2015.7158165 (cit. on p. 91).
 - [96] G. Grassi et al. "ACM HotMobile 2013 poster: vehicular inter-networking via named data". In: *ACM SIGMOBILE Mobile Computing and Communications Review* 17 (2013-11), pp. 23–24. doi: 10.1145/2542095.2542108 (cit. on pp. 40, 55, 56).
 - [97] G. Grassi et al. "VANET via Named Data Networking". In: *2014 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*. 2014-04, pp. 410–415. doi: 10.1109/INFOCOMW.2014.6849267 (cit. on pp. 5, 25, 26, 38, 40, 43, 52, 55, 59, 66, 73, 76, 80, 83, 90, 91, 95, 96, 101, 121).
 - [98] D. Grewe et al. "ADePt: Adaptive distributed content prefetching for information-centric connected vehicles". In: vol. 2018-June. 2018, pp. 1–5. doi: 10.1109/VTCSpring.2018.8417748 (cit. on pp. 44, 52, 54, 68).

- [99] D. Grewe et al. "Caching-as-a-Service in Virtualized Caches for Information-Centric Connected Vehicle Environments". In: vol. 2018-December. 2019. doi: 10.1109/VNC.2018.8628354 (cit. on pp. 44, 52, 54, 68, 72).
- [100] D. Grewe, M. Wagner, and H. Frey. "PeRCeIVE: Proactive caching in ICN-based VANETs". In: *2016 IEEE Vehicular Networking Conference (VNC)*. 2016-12, pp. 1–8. doi: 10.1109/VNC.2016.7835962 (cit. on pp. 43, 51, 52, 65, 66).
- [101] J. Gu et al. "Node Value and Content Popularity-Based Caching Strategy for Massive VANETs". In: *Wireless Communications and Mobile Computing 2021 (2021)*. doi: 10.1155/2021/5585896 (cit. on pp. 45, 52, 55, 69).
- [102] H. Guo et al. "A New ICN routing selecting algorithm based on Link Expiration Time of VANET under the highway environment". In: *2017 IFIP/IEEE Symposium on Integrated Network and Service Management (IM)*. 2017-05, pp. 640–643. doi: 10.23919/INM.2017.7987349 (cit. on pp. 44, 47, 48, 62).
- [103] D. Gupta et al. "ICN-Based Enhanced Cooperative Caching for Multimedia Streaming in Resource Constrained Vehicular Environment". In: *IEEE Transactions on Intelligent Transportation Systems* 22.7 (2021), pp. 4588–4600. doi: 10.1109/TITS.2020.3043593 (cit. on pp. 45, 52, 53, 65, 66, 69).
- [104] J. Haerri, C. Bonnet, and F. Filali. *The challenges of predicting mobility*. Tech. rep. EURECOM+2240. Eurecom, 2006-08. url: <http://www.eurecom.fr/publication/2240> (cit. on p. 27).
- [105] H. Hartenstein and L. P. Laberteaux. "A tutorial survey on vehicular ad hoc networks". In: *IEEE Communications Magazine* 46.6 (2008-06), pp. 164–171. issn: 1558-1896. doi: 10.1109/MCOM.2008.4539481 (cit. on pp. 15, 17).
- [106] K. Hasan and S.-H. Jeong. "CCN-based vehicular communications". In: *2018 International Conference on Information Networking (ICOIN)*. 2018-01, pp. 926–929. doi: 10.1109/ICOIN.2018.8343259 (cit. on pp. 44, 52, 54, 68).
- [107] K. Hasan and S.-H. Jeong. "Proactive Caching in the Vehicular Networks based on Content Locality". In: *2020 International Conference on Information and Communication Technology Convergence (ICTC)*. 2020-10, pp. 170–172. doi: 10.1109/ICTC49870.2020.9289251 (cit. on pp. 45, 52, 54, 68).
- [108] Y. He et al. "Resource Allocation in Software-Defined and Information-Centric Vehicular Networks with Mobile Edge Computing". In: *2017 IEEE 86th Vehicular Technology Conference (VTC-Fall)*. 2017-09, pp. 1–5. doi: 10.1109/VTCFall.2017.8288203 (cit. on pp. 44, 52).
- [109] L. Hou et al. "A Q-Learning-Based Proactive Caching Strategy for Non-Safety Related Services in Vehicular Networks". In: *IEEE Internet of Things Journal* 6.3 (2019), pp. 4512–4520 (cit. on pp. 44, 52, 53, 65, 68).

-
- [110] W. Huang et al. "Cluster-Based Cooperative Caching With Mobility Prediction in Vehicular Named Data Networking". In: *IEEE Access* 7 (2019), pp. 23442–23458. issn: 2169-3536. doi: 10.1109/ACCESS.2019.2897747 (cit. on pp. 44, 52, 53, 65, 68).
 - [111] W. Huang et al. "Cluster-based Selective Cooperative Caching Strategy in Vehicular Named Data Networking". In: *2018 1st IEEE International Conference on Hot Information-Centric Networking (HotICN)*. 2018-08, pp. 7–12. doi: 10.1109/HOTICN.2018.8605950 (cit. on pp. 44, 52, 53, 68).
 - [112] R. Hussain et al. "Realization of VANET-Based Cloud Services through Named Data Networking". In: *IEEE Communications Magazine* 56.8 (2018-08), pp. 168–175. issn: 1558-1896. doi: 10.1109/MCOM.2018.1700514 (cit. on pp. 2, 6).
 - [113] M. Ibnkahla. "Applications of neural networks to digital communications-survey". In: *Signal Processing* 80 (2000-07), pp. 1185–1215. doi: 10.1016/S0165-1684(00)00030-X (cit. on p. 28).
 - [114] ICN Research Group. *Charter for Research Group*. 2020-02. url: <https://datatracker.ietf.org/rg/icnrg/about/> (cit. on p. 40).
 - [115] IEEE. *IEEE Xplore*. 2020-03. url: <https://ieeexplore.ieee.org/Xplore/home.jsp> (cit. on p. 41).
 - [116] A. Ioannou and S. Weber. "A Survey of Caching Policies and Forwarding Mechanisms in Information-Centric Networking". In: *IEEE Communications Surveys Tutorials* 18.4 (2016), pp. 2847–2886. doi: 10.1109/COMST.2016.2565541 (cit. on p. 38).
 - [117] N. G. P. N. E. I. S. G. (ISG). *Next Generation Protocols (NGP); An example of a non-IP network protocol architecture based on RINA design principles*. Tech Report. European Telecommunications Standards Institute (ETSI), 2019 (cit. on pp. 4, 20).
 - [118] V. Jacobson et al. "Networking Named Content". In: *Proceedings of the 5th International Conference on Emerging Networking Experiments and Technologies*. CoNEXT '09. Rome, Italy: ACM, 2009, pp. 1–12. isbn: 978-1-60558-636-6. doi: 10.1145/1658939.1658941 (cit. on pp. 2, 19).
 - [119] Z. u. A. Jaffri, Z. Ahmad, and M. Tahir. "Named Data Networking (NDN), New Approach to Future Internet Architecture Design: A Survey". In: *International Journal of Informatics and Communication Technology (IJ-ICT)* 2 (2013-12). doi: 10.11591/ij-ict.v2i3.5122 (cit. on p. 23).
 - [120] J. Jeong. *IPv6 Wireless Access in Vehicular Environments (IPWAVE): Problem Statement and Use Cases*. Internet-Draft draft-ietf-ipwave-vehicular-networking-13. IETF Secretariat, 2020-01. url: <http://www.ietf.org/internet-drafts/draft-ietf-ipwave-vehicular-networking-13.txt> (cit. on p. 14).

- [121] S. Julier and J. Uhlmann. "New extension of the Kalman filter to nonlinear systems". In: *Defense, Security, and Sensing*. 1997 (cit. on p. 28).
- [122] J. Junior and E. Wille. "Routing in Vehicular Ad Hoc Networks: Main Characteristics and Tendencies". In: *Journal of Computer Networks and Communications* (2018-03). issn: 2090-7141. doi: 10.1155/2018/13021232 (cit. on pp. 46, 59, 98).
- [123] A. Kaci and A. Rachedi. "Named data networking architecture for internet of vehicles in the era of 5G". In: *Annals of Telecommunications* 76 (2021-07), pp. 1–13. doi: 10.1007/s12243-021-00866-8 (cit. on pp. 45, 52, 56, 69).
- [124] R. E. Kalman. "A New Approach to Linear Filtering and Prediction Problems". In: *Journal of Basic Engineering* 82.1 (1960-03), pp. 35–45. issn: 0021-9223. doi: 10.1115/1.3662552 (cit. on p. 28).
- [125] E. Kalogeiton and T. Braun. "Infrastructure-Assisted Communication for NDN-VANETs". In: *2018 IEEE 19th International Symposium on "A World of Wireless, Mobile and Multimedia Networks" (WoWMoM)*. 2018-06, pp. 1–10. doi: 10.1109/WoWMoM.2018.8449740 (cit. on pp. 44, 46, 47, 62, 94, 99).
- [126] E. Kalogeiton, D. Iapello, and T. Braun. "A Geographical Aware Routing Protocol Using Directional Antennas for NDN-VANETs". In: *2019 IEEE 44th Conference on Local Computer Networks (LCN)*. 2019-10, pp. 133–136. doi: 10.1109/LCN44214.2019.8990760 (cit. on pp. 44, 46, 47, 62).
- [127] E. Kalogeiton, D. Iapello, and T. Braun. "Equipping NDN-VANETs with Directional Antennas for Efficient Content Retrieval". In: *2020 IEEE 17th Annual Consumer Communications Networking Conference (CCNC)*. 2020, pp. 1–8 (cit. on pp. 44, 46, 47, 63).
- [128] E. Kalogeiton, T. Kolonko, and T. Braun. "A multihop and multipath routing protocol using NDN for VANETs". In: *2017 16th Annual Mediterranean Ad Hoc Networking Workshop (Med-Hoc-Net)*. 2017-06, pp. 1–8. doi: 10.1109/MedHocNet.2017.8001640 (cit. on pp. 43, 46, 47, 62, 99).
- [129] E. Kalogeiton, T. Kolonko, and T. Braun. "A topology-oblivious routing protocol for NDN-VANETs". In: *Annals of Telecommunications* (2017-11). doi: 10.1007/s12243-018-0661-4 (cit. on pp. 43, 46, 47, 62, 97, 99).
- [130] E. Kalogeiton, Z. Zhao, and T. Braun. "Is SDN the solution for NDN-VANETs?" In: *2017 16th Annual Mediterranean Ad Hoc Networking Workshop (Med-Hoc-Net)*. 2017-06, pp. 1–6. doi: 10.1109/MedHocNet.2017.8001643 (cit. on pp. 43, 47).
- [131] J. Kangasharju, J. Ott, and O. Karkulahti. "Floating content: Information availability in urban environments". In: *2010 8th IEEE International Conference on Pervasive Computing and Communications Workshops (PERCOM Workshops)*. 2010, pp. 804–808. doi: 10.1109/PERCOMW.2010.5470546 (cit. on p. 72).

-
- [132] G. Karagiannis et al. "Vehicular Networking: A Survey and Tutorial on Requirements, Architectures, Challenges, Standards and Solutions". In: *IEEE Communications Surveys Tutorials* 13.4 (2011-04), pp. 584–616. issn: 2373-745X. doi: 10.1109/SURV.2011.061411.00019 (cit. on p. 17).
 - [133] T. Kato et al. "How to Implement NDN MANET over ndnSIM Simulator". In: *2018 IEEE 4th International Conference on Computer and Communications (ICCC)*. 2018-12, pp. 451–456. doi: 10.1109/CompComm.2018.8780792 (cit. on pp. 99, 111).
 - [134] H. Kawashima. "Japanese perspective of driver information systems". In: *Transportation* 17 (1990), pp. 263–284 (cit. on p. 1).
 - [135] C. Kerrche et al. "Internet of Vehicles Over Named Data Networking: Current Status and Future Challenges". In: 2019-07, <https://www.springer.com/us/book/9783030227722>. isbn: 978-3-030-22773-9. doi: 10.1007/978-3-030-22773-9_7 (cit. on pp. 38, 39, 83).
 - [136] J. A. Khan and Y. Ghamri-Doudane. "STRIVE: Socially-Aware Three-Tier Routing in Information-Centric Vehicular Environment". In: *2016 IEEE Global Communications Conference (GLOBECOM)*. 2016-12, pp. 1–7. doi: 10.1109/GLOCOM.2016.7842301 (cit. on pp. 43, 47, 48, 62).
 - [137] J. Khan and Y. Ghamri-Doudane. "Car Rank: An Information-Centric Identification of Important Smart Vehicles for Urban Sensing". In: *2015 IEEE 14th International Symposium on Network Computing and Applications*. 2015, pp. 184–191 (cit. on p. 48).
 - [138] J. Khan, Y. Ghamri-Doudane, and D. Botvich. "GRank - An Information-Centric Autonomous and Distributed Ranking of Popular Smart Vehicles". In: *2015 IEEE Global Communications Conference (GLOBECOM)*. 2015, pp. 1–7 (cit. on p. 48).
 - [139] F. Khandaker et al. "A functional taxonomy of caching schemes: Towards guided designs in information-centric networks". In: *Comput. Networks* 165 (2019) (cit. on p. 51).
 - [140] H. Khelifi et al. "Named Data Networking in Vehicular Ad Hoc Networks: State-of-the-Art and Challenges". In: *IEEE Communications Surveys Tutorials* 22.1 (2019-01), pp. 320–351. issn: 2373-745X. doi: 10.1109/COMST.2019.2894816 (cit. on pp. 2, 5, 6, 17, 20, 25, 26, 38, 39, 51, 73, 84).
 - [141] H. Khelifi et al. "A QoS-Aware Cache Replacement Policy for Vehicular Named Data Networks". In: *2019 IEEE Global Communications Conference (GLOBECOM)*. 2019, pp. 1–6 (cit. on pp. 44, 52, 56, 64).
 - [142] H. Khelifi et al. "An Optimized Proactive Caching Scheme Based on Mobility Prediction for Vehicular Networks". In: *2018 IEEE Global Communications Conference (GLOBECOM)*. 2018-12, pp. 1–6. doi: 10.1109/GLOCOM.2018.8647898 (cit. on pp. 44, 52, 53, 65, 67).

- [143] H. Khelifi et al. "An Optimized Proactive Caching Scheme Based on Mobility Prediction for Vehicular Networks". In: *2018 IEEE Global Communications Conference (GLOBECOM)*. 2018-12, pp. 1–6. doi: 10.1109/GLOCOM.2018.8647898 (cit. on pp. 57, 58).
- [144] H. Khelifi et al. "In-Network Caching in ICN-based Vehicular Networks: Effectiveness and Performance Evaluation". In: *ICC 2020 - 2020 IEEE International Conference on Communications (ICC)*. 2020, pp. 1–6. doi: 10.1109/ICC40277.2020.9148950 (cit. on p. 10).
- [145] H. Khelifi et al. "Reputation-Based Blockchain for Secure NDN Caching in Vehicular Networks". In: *2018 IEEE Conference on Standards for Communications and Networking (CSCN)*. 2018, pp. 1–6. doi: 10.1109/CSCN.2018.8581849 (cit. on p. 44).
- [146] W. Khreich et al. "On the Memory Complexity of the Forward-Backward Algorithm". In: *Pattern Recogn. Lett.* 31.2 (2010-01), pp. 91–99. issn: 0167-8655. doi: 10.1016/j.patrec.2009.09.023 (cit. on p. 34).
- [147] B. Kitchenham and S. Charters. "Guidelines for performing Systematic Literature Reviews in Software Engineering". In: 2 (2007-01) (cit. on p. 40).
- [148] T. Koponen et al. "A Data-oriented (and Beyond) Network Architecture". In: *Proceedings of the 2007 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications*. SIGCOMM '07. Kyoto, Japan: ACM, 2007, pp. 181–192. isbn: 978-1-59593-713-1. doi: 10.1145/1282380.1282402 (cit. on p. 19).
- [149] M. Kuai and X. Hong. "Location-Based Deferred Broadcast for Ad-Hoc Named Data Networking". In: *Future Internet* 11.6 (2019). issn: 1999-5903. doi: 10.3390/fi11060139 (cit. on pp. 76, 80).
- [150] M. Kuai, X. Hong, and Q. Yu. "Density-Aware Delay-Tolerant Interest Forwarding in Vehicular Named Data Networking". In: *2016 IEEE 84th Vehicular Technology Conference (VTC-Fall)*. 2016, pp. 1–5. doi: 10.1109/VTCFall.2016.7880953 (cit. on pp. 76, 79, 80).
- [151] D. Kutscher et al. *Information-Centric Networking (ICN) Research Challenges*. RFC 7927. 2016-07. doi: 10.17487/RFC7927 (cit. on pp. 2, 20, 40).
- [152] D. B. Lange and M. Oshima. "Seven Good Reasons for Mobile Agents". In: *Commun. ACM* 42.3 (1999-03), pp. 88–89. issn: 0001-0782. doi: 10.1145/295685.298136 (cit. on p. 63).
- [153] K. Lei et al. "Blockchain-Based Cache Poisoning Security Protection and Privacy-Aware Access Control in NDN Vehicular Edge Computing Networks". In: *Journal of Grid Computing* (2020), pp. 1–21 (cit. on pp. 45, 57, 58).
- [154] D. Li et al. "A reliable and efficient forwarding strategy in vehicular named data networking". In: *Journal of Communications and Networks* 22.4 (2020), pp. 348–357. doi: 10.1109/JCN.2020.0000014 (cit. on pp. 73, 75, 79, 80).

- [155] Y. Li et al. "Limits of Predictability for Large-Scale Urban Vehicular Mobility". In: *IEEE Transactions on Intelligent Transportation Systems* 15.6 (2014), pp. 2671–2682. doi: 10.1109/TITS.2014.2325395 (cit. on pp. 26, 101).
- [156] Y. Li et al. "Context-Aware Data Dissemination for ICN-Based Vehicular Ad Hoc Networks". In: *Information* 9.11 (2018). issn: 2078-2489. doi: 10.3390/info9110263 (cit. on pp. 76, 79, 80).
- [157] L. Liu et al. "CCN-based cooperative caching in VANET". In: *2015 International Conference on Connected Vehicles and Expo (ICCVE)*. 2015, pp. 198–203 (cit. on pp. 43, 52, 53, 65, 66).
- [158] S. Liu et al. *A Driving Intention Prediction Method Based on Hidden Markov Model for Autonomous Driving*. 2019. arXiv: 1902.09068 [cs.LG] (cit. on pp. 44, 57, 59, 70).
- [159] X. Liu et al. "Information-centric mobile ad hoc networks and content routing: A survey". In: *Ad Hoc Networks* 58 (2017). Hybrid Wireless Ad Hoc Networks, pp. 255 –268. issn: 1570-8705. doi: <https://doi.org/10.1016/j.adhoc.2016.04.005> (cit. on pp. 38, 39).
- [160] X. Liu et al. "A Geographic Opportunistic Forwarding Strategy for Vehicular Named Data Networking". In: *Intelligent Distributed Computing IX*. Ed. by P. Novais et al. Cham: Springer International Publishing, 2016, pp. 509–521. isbn: 978-3-319-25017-5 (cit. on pp. 73, 74, 79, 80).
- [161] P. A. Lopez et al. "Microscopic Traffic Simulation using SUMO". In: *2019 IEEE Intelligent Transportation Systems Conference (ITSC)*. IEEE, 2018-11, pp. 2575–2582. url: <https://elib.dlr.de/127994/> (cit. on pp. 106, 108, 122).
- [162] J. M. Lourenço. *The NOVAthesis L^AT_EX Template User's Manual*. NOVA University Lisbon. 2021. url: <https://github.com/joaomlourenco/novathesis/raw/main/template.pdf> (cit. on p. ii).
- [163] S. M. Zaki et al. "A Location Based Prediction Service Protocol for VANET City Environment". In: *International Journal of Innovative Computing, Information and Control (IJICIC), Indexed in ISI with 1.668 impact factor 8* (2012-10), pp. 1–26 (cit. on p. 28).
- [164] L. Ma et al. "Distributed Probabilistic Caching with Content-Location Awareness in VNDNs". In: *Proceedings of the 21st ACM International Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems*. MSWIM '18. Montreal, QC, Canada: Association for Computing Machinery, 2018, pp. 311–314. isbn: 9781450359603. doi: 10.1145/3242102.3242146 (cit. on pp. 44, 52, 55, 67).
- [165] S. Madanapalli et al. "ROOF (Realtime Onsite Operations Facilitation) Computing Framework". In: *ROOF (Realtime Onsite Operations Facilitation) Computing Framework* (2022), pp. 1–48 (cit. on p. 56).
- [166] M. F. Majeed, S. H. Ahmed, and M. N. Dailey. "Enabling Push-Based Critical Data Forwarding in Vehicular Named Data Networks". In: *IEEE Communications Letters* 21.4 (2017), pp. 873–876 (cit. on p. 90).

- [167] A. Malhi, S. Batra, and H. Pannu. "Security of Vehicular Ad-hoc Networks: A Comprehensive Survey". In: *Computers and Security* 89 (2019-11), p. 101664. doi: 10.1016/j.cose.2019.101664 (cit. on p. 17).
- [168] P. Manimaran and A. R. K. P. "NDNIDS: An Intrusion Detection System for NDN Based VANET". In: *2020 IEEE 91st Vehicular Technology Conference (VTC2020-Spring)*. 2020-05, pp. 1–5. doi: 10.1109/VTC2020-Spring48590.2020.9129365 (cit. on pp. 45, 57, 58, 70).
- [169] F. J. Martinez et al. "Computer Simulations of VANETs Using Realistic City Topologies". In: *Wireless Personal Communications* 69 (2013), pp. 639–663. issn: 1572-834X. doi: 10.1007/s11277-012-0594-6 (cit. on p. 87).
- [170] H. Maryam et al. "TFS: A reliable routing protocol for Vehicular Content Centric Networks". In: *2017 13th International Conference on Emerging Technologies (ICET)*. 2017, pp. 1–6. doi: 10.1109/ICET.2017.8281740 (cit. on pp. 44, 47, 50, 62).
- [171] S. Mastorakis, A. Afanasyev, and L. Zhang. "On the Evolution of ndnSIM: an Open-Source Simulator for NDN Experimentation". In: *ACM Computer Communication Review* (2017-07) (cit. on pp. 106, 107, 122).
- [172] S. Mastorakis et al. "nTorrent: Peer-to-Peer File Sharing in Named Data Networking". In: *2017 26th International Conference on Computer Communication and Networks (ICCCN)*. 2017, pp. 1–10. doi: 10.1109/ICCCN.2017.8038462 (cit. on p. 2).
- [173] W. Mathew, R. Raposo, and B. Martins. "Predicting Future Locations with Hidden Markov Models". In: *Proceedings of the 2012 ACM Conference on Ubiquitous Computing*. UbiComp '12. Pittsburgh, Pennsylvania: Association for Computing Machinery, 2012, pp. 911–918. isbn: 9781450312240. doi: 10.1145/2370216.2370421 (cit. on p. 103).
- [174] G. Mauri et al. "Optimal Content Prefetching in NDN Vehicle-to-Infrastructure Scenario". In: *IEEE Transactions on Vehicular Technology* 66.3 (2016-03), pp. 2513–2525. issn: 1939-9359. doi: 10.1109/TVT.2016.2580586 (cit. on pp. 43, 51, 52, 65, 67).
- [175] S. Meyn and R. Tweedie. *Markov Chains and Stochastic Stability*. 2009-01. isbn: 978-0-521-73182-9. doi: 10.1007/978-1-4471-3267-7 (cit. on p. 27).
- [176] T. Mikami. "CACS - Urban traffic control system featuring computer control". In: *Managing Requirements Knowledge, International Workshop on*. Los Alamitos, CA, USA: IEEE Computer Society, 1978-06, p. 1265. doi: 10.1109/AFIPS.1978.38 (cit. on p. 1).
- [177] S. Misra, I. Woungang, and S. C. Misra. *Guide to Wireless Ad Hoc Networks*. 2009th. Springer Publishing Company, Incorporated, 2009. isbn: 1848003277 (cit. on pp. 3, 13, 14, 98).

- [178] T. Miyazaki, H. Shigeno, and T. Yashiro. "Routing Protocol Using Virtual Nodes for Traffic Information Collection in Information Centric Vehicular Ad-hoc Networks". In: *2019 Twelfth International Conference on Mobile Computing and Ubiquitous Network (ICMU)*. 2019, pp. 1–6. doi: 10.23919/ICMU48249.2019.9006670 (cit. on pp. 44, 47, 50, 63, 72).
- [179] F. Modesto and A. Boukerche. "Social Pre-caching for Location-dependent Requests in Vehicular Information-Centric Ad-hoc Networks". In: *2018 IEEE Symposium on Computers and Communications (ISCC)*. 2018-06, pp. 00426–00431. doi: 10.1109/ISCC.2018.8538613 (cit. on pp. 44, 51, 52, 66, 67).
- [180] F. Modesto and A. Boukerche. "Utility-Gradient Implicit Cache Coordination Policy for Information-Centric Ad-Hoc Vehicular Networks". In: *2017 IEEE 42nd Conference on Local Computer Networks (LCN)*. 2017-10, pp. 10–17. doi: 10.1109/LCN.2017.61 (cit. on pp. 44, 52, 54, 67).
- [181] B. Mokhtar and M. Azab. "Survey on Security Issues in Vehicular Ad Hoc Networks". In: *Alexandria Engineering Journal* 54.4 (2015), pp. 1115–1126. issn: 1110-0168. doi: <https://doi.org/10.1016/j.aej.2015.07.011> (cit. on p. 57).
- [182] T. Nakazawa, S. Obana, and S. Tang. "Efficient Collection of Road and Traffic Information by CCN-based Inter-Vehicle Communications". In: *2018 24th Asia-Pacific Conference on Communications (APCC)*. 2018, pp. 286–291. doi: 10.1109/APCC.2018.8633556 (cit. on pp. 44, 47, 48, 62).
- [183] Y. Nam et al. "Adaptive content precaching scheme based on the predictive speed of vehicles in content-centric vehicular networks". In: *Sensors* 21.16 (2021). doi: 10.3390/s21165376 (cit. on pp. 2, 45, 52, 53, 65, 69).
- [184] ndnSIM. *NS-3 based Named Data Networking (NDN) simulator*. 2022-10. url: <https://ndnsim.net/2.7/intro.html> (cit. on p. 106).
- [185] E. Newberry, X. Ma, and L. Zhang. "YaNFD: Yet another named data networking forwarding daemon". In: *Proceedings of the 8th ACM Conference on Information-Centric Networking*. ICN '21. Paris, France: Association for Computing Machinery, 2021, pp. 30–41. isbn: 9781450384605. doi: 10.1145/3460417.3482969 (cit. on p. 2).
- [186] M. NGO et al. "A Visual-Identification Based Forwarding Strategy for Vehicular Named Data Networking". In: *IEICE Transactions on Information and Systems* E106.D.2 (2023), pp. 204–217. doi: 10.1587/transinf.2022EDP7084 (cit. on pp. 74, 80).
- [187] T. Nguyen et al. "Reliable Detection of Interest Flooding Attack in Real Deployment of Named Data Networking". In: *IEEE Transactions on Information Forensics and Security* 14.9 (2019), pp. 2470–2485. doi: 10.1109/TIFS.2019.2899247 (cit. on p. 57).
- [188] NS-3. *NS-3: Network simulator*. 2022-10. url: <https://www.nsnam.org/> (cit. on pp. 106, 108, 122).

- [189] J. F. Nunamaker, M. Chen, and T. D. M. Purdin. "Systems Development in Information Systems Research". In: *Journal of Management Information Systems* 7.3 (1990), pp. 89–106. issn: 07421222. (Visited on 2025-04-28) (cit. on p. 8).
- [190] S. Ostrovskaya et al. "Towards Multi-metric Cache Replacement Policies in Vehicular Named Data Networks". In: *2018 IEEE 29th Annual International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC)*. 2018-09, pp. 1–7. doi: 10.1109/PIMRC.2018.8580741 (cit. on pp. 44, 52, 56, 64).
- [191] D. Parekh et al. "A Review on Autonomous Vehicles: Progress, Methods and Challenges". In: *Electronics* 11.14 (2022). issn: 2079-9292. doi: 10.3390/electronics11142162 (cit. on p. 3).
- [192] C.-M. Park et al. "Enhanced Protocol for Wireless Content-Centric Network". In: vol. 6. 2016-05. doi: 10.5121/csit.2016.60614 (cit. on p. 99).
- [193] S. Park et al. "Mobility-aware Distributed Proactive Caching in Content-Centric Vehicular Networks". In: *2019 12th IFIP Wireless and Mobile Networking Conference (WMNC)*. 2019, pp. 175–180 (cit. on pp. 44, 52, 53, 65, 68).
- [194] V. Patil et al. "Decentralized Photo Sharing via Named Data Networking". In: *Proceedings of the 10th ACM Conference on Information-Centric Networking*. ACM ICN '23. Reykjavik, Iceland: Association for Computing Machinery, 2023, pp. 118–120. isbn: 9798400704031. doi: 10.1145/3623565.3623755 (cit. on p. 2).
- [195] T. Pavithra and B. Nagabhushana. "A Survey on Security in VANETs". In: *2020 Second International Conference on Inventive Research in Computing Applications (ICIRCA)*. 2020, pp. 881–889. doi: 10.1109/ICIRCA48905.2020.9182823 (cit. on p. 57).
- [196] K. Peffers et al. "A design science research methodology for information systems research". In: *Journal of Management Information Systems* 24 (2007-01), pp. 45–77 (cit. on p. 8).
- [197] K. Pentikousis et al. *Information-Centric Networking: Baseline Scenarios*. RFC 7476. RFC Editor, 2015-03 (cit. on pp. 40, 60, 64).
- [198] A. Perallos et al. *Intelligent Transport Systems: Technologies and Applications*. Wiley, 2015-11. isbn: 978-1-118-89478-1 (cit. on pp. 1, 14).
- [199] A. Pereira et al. "Named Data for Mobile AdHoc Networks". In: *Intelligent Environments* (2016). doi: 10.3233/978-1-61499-690-3-28 (cit. on p. 5).
- [200] K. Plossl, T. Nowey, and C. Mletzko. "Towards a security architecture for vehicular ad hoc networks". In: 2006-05, 8 pp. isbn: 0-7695-2567-9. doi: 10.1109/ARES.2006.136 (cit. on pp. 6, 7, 97, 98, 137).

-
- [201] A. A. Prates, I. V. Bastos, and I. M. Moraes. "GeoZone: An interest-packet forwarding mechanism based on dissemination zone for content-centric vehicular networks". In: *Computers and Electrical Engineering* 73 (2019), pp. 155–166. issn: 0045-7906. doi: 10.1016/j.compeleceng.2018.11.015 (cit. on pp. 78, 80).
- [202] N. D. N. N. project. *NDN Packet Format Specification version 0.3*. url: <https://named-data.net/doc/NDN-packet-spec/current/> (cit. on p. 21).
- [203] ndn-cxx Project Team. *ndn-cxx: NDN C++ library with eXperimental eXtensions*. 2022-11. url: <https://named-data.net/doc/ndn-cxx/current/INSTALL.html> (cit. on p. 106).
- [204] W. Quan et al. "Intelligent popularity-aware content caching and retrieving in highway vehicular networks". In: *Eurasip Journal on Wireless Communications and Networking* 2016.1 (2016). doi: 10.1186/s13638-016-0688-z (cit. on pp. 43, 52, 65, 67).
- [205] W. Quan et al. "Social cooperation for information-centric multimedia streaming in highway VANETs". In: *Proceeding of IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks 2014*. 2014-06, pp. 1–6. doi: 10.1109/WoWMoM.2014.6918992 (cit. on pp. 43, 52, 54, 65, 66).
- [206] J. Rabari and A. R. P. Kumar. "FIFA: Fighting against Interest Flooding Attack in NDN-based VANET". In: *2021 International Wireless Communications and Mobile Computing (IWCMC)*. 2021-06, pp. 1539–1544. doi: 10.1109/IWCMC51323.2021.9498767 (cit. on pp. 45, 57, 58).
- [207] L. R. Rabiner. "A tutorial on hidden Markov models and selected applications in speech recognition". In: *Proceedings of the IEEE* 77.2 (1989), pp. 257–286. doi: 10.1109/5.18626 (cit. on pp. 27, 32–34, 101).
- [208] B. Rainer and S. Petscharnig. "Challenges and Opportunities of Named Data Networking in Vehicle-To-Everything Communication: A Review". In: *Information* 9.11 (2018). issn: 2078-2489. doi: 10.3390/info9110264 (cit. on pp. 17, 25, 83).
- [209] N. Rao and R. Sharma. "Survey of Content Naming Schemes in Vehicular Named Data Network". In: 2021-05, pp. 1–6. doi: 10.1109/INCET51464.2021.9456419 (cit. on p. 39).
- [210] M. Raya, P. Papadimitratos, and J.-p. Hubaux. "SECURING VEHICULAR COMMUNICATIONS". In: *IEEE Wireless Communications* 13.5 (2006), pp. 8–15. doi: 10.1109/WC-M.2006.250352 (cit. on p. 17).
- [211] J. Rexford and C. Dovrolis. "Future Internet Architecture: Clean-slate Versus Evolutionary Research". In: *Commun. ACM* 53.9 (2010-09), pp. 36–40. issn: 0001-0782. doi: 10.1145/1810891.1810906 (cit. on pp. 2, 4).

- [212] L. B. Rondon et al. "A Distance and Position-based Caching Discovery Protocol for Vehicular Named-Data Networks". In: *2019 IEEE Latin-American Conference on Communications (LATIN-COM)*. 2019, pp. 1–6. doi: 10.1109/LATINCOM48065.2019.8938022 (cit. on pp. 76, 77, 80).
- [213] A. Rowstron and G. Pau. "Characteristics of a vehicular network". In: *University of California Los Angeles, Computer Science Department, Tech. Rep.* 2009-07. url: <https://www.microsoft.com/en-us/research/publication/characteristics-vehicular-network/> (cit. on pp. 91, 95).
- [214] L. Rui et al. "MUPF: Multiple unicast path forwarding in content-centric VANETs". In: *Ad Hoc Networks* 81 (2018), pp. 211–225. doi: 10.1016/j.adhoc.2018.06.012 (cit. on pp. 44, 47, 62, 97).
- [215] Z. Sabir and A. Amine. "Connected Vehicles using NDN: Security Concerns and Remaining Challenges". In: *2021 7th International Conference on Optimization and Applications (ICOA)*. 2021-05, pp. 1–6. doi: 10.1109/ICOA51614.2021.9442640 (cit. on p. 57).
- [216] M. Saini, A. Alelaiwi, and A. E. Saddik. "How Close Are We to Realizing a Pragmatic VANET Solution? A Meta-Survey". In: *ACM Comput. Surv.* 48.2 (2015-11). issn: 0360-0300. doi: 10.1145/2817552 (cit. on pp. 3, 17, 25).
- [217] M. U. Sattar and R. A. Rehman. "Interest Flooding Attack Mitigation in Named Data Networking Based VANETs". In: *2019 International Conference on Frontiers of Information Technology (FIT)*. 2019, pp. 245–2454. doi: 10.1109/FIT47737.2019.00053 (cit. on pp. 44, 57, 58).
- [218] A. H. Sayed, P. M. Djuric, and F. Hlawatsch. "Chapter 6 - Distributed Kalman and Particle Filtering". In: *Cooperative and Graph Signal Processing*. Ed. by P. M. Djuric and C. Richard. Academic Press, 2018, pp. 169 –207. isbn: 978-0-12-813677-5. doi: 10.1016/B978-0-12-813677-5.00006-7 (cit. on p. 28).
- [219] E. Schoch et al. "Communication patterns in VANETs". In: *IEEE Communications Magazine* 46.11 (2008), pp. 119–125. doi: 10.1109/MCOM.2008.4689254 (cit. on p. 14).
- [220] M. Schütz and K. Dietmayer. "A Flexible Environment Perception Framework for Advanced Driver Assistance Systems". In: 2013-01, pp. 21–29. doi: 10.1007/978-3-319-00476-1_3 (cit. on pp. 7, 14).
- [221] Scopus. *Scopus*. 2020-03. url: <https://www2.scopus.com/> (cit. on p. 41).
- [222] J. Seedorf et al. *Research Directions for Using ICN in Disaster Scenarios*. Internet-Draft draft-irtf-icnrg-disaster-10. Work in Progress. Internet Engineering Task Force, 2020-01. 20 pp. url: <https://datatracker.ietf.org/doc/html/draft-irtf-icnrg-disaster-10> (cit. on p. 40).

- [223] O. Senouci, A. Zibouda, and S. Harous. "Survey: Routing Protocols in Vehicular Ad Hoc Networks". In: *Proceedings of the Second International Conference on Advanced Wireless Information, Data, and Communication Technologies*. AWICT 2017. Paris, France: Association for Computing Machinery, 2017. isbn: 9781450353106. doi: 10.1145/3231830.3231838 (cit. on p. 98).
- [224] M. S. Sheikh, J. Liang, and W. Wang. "A Survey of Security Services, Attacks, and Applications for Vehicular Ad Hoc Networks (VANETs)". In: *Sensors* 19.16 (2019). issn: 1424-8220. doi: 10.3390/s19163589 (cit. on p. 57).
- [225] I. Shemsi and P. Kadam. "Named Data Networking in VANET: A Survey". In: 2017 (cit. on p. 39).
- [226] J. Shi, E. Newberry, and B. Zhang. "On broadcast-based self-learning in named data networking". In: *2017 IFIP Networking Conference (IFIP Networking) and Workshops*. 2017, pp. 1–9. doi: 10.23919/IFIPNetworking.2017.8264832 (cit. on p. 24).
- [227] J. Shi, D. Pesavento, and L. Benmohamed. "NDN-DPDK: NDN Forwarding at 100 Gbps on Commodity Hardware". In: *Proceedings of the 7th ACM Conference on Information-Centric Networking*. ICN '20. Virtual Event, Canada: Association for Computing Machinery, 2020, pp. 30–40. isbn: 9781450380409. doi: 10.1145/3405656.3418715 (cit. on p. 2).
- [228] A. Siddiqua et al. "Optimal Routing Protocol For Efficient Communication in a Social Vehicular Content-Centric Network". In: *2021 IEEE 94th Vehicular Technology Conference (VTC2021-Fall)*. 2021, pp. 1–5. doi: 10.1109/VTC2021-Fall52928.2021.9625421 (cit. on pp. 45, 47, 49, 63).
- [229] S. Signorello, M. R. Palattella, and L. A. Grieco. "Security Challenges in Future NDN-Enabled VANETs". In: *2016 IEEE Trustcom/BigDataSE/ISPA*. 2016, pp. 1771–1775. doi: 10.1109/TrustCom.2016.0272 (cit. on pp. 38, 57).
- [230] E. T. d. Silva, J. M. H. Macedo, and A. L. D. Costa. "Context-Aware Routing and Forwarding Model for NDN-Based VANET". In: *Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering*. Vol. 428. 2022, pp. 1–15. doi: 10.1007/978-3-030-98005-4_2 (cit. on pp. 6, 9, 10, 45, 47, 48, 60, 63, 83).
- [231] E. T. da Silva. *Routing in Named Data Networking - Literature review*. Tech Report. Algoritmi Centre - School of Engineering - University of Minho, 2019-04 (cit. on p. 8).
- [232] E. T. da Silva, A. L. D. Costa, and J. M. H. de Macedo. "On the realization of VANET using named data networking: On improvement of VANET using NDN-based routing, caching, and security". In: *International Journal of Communication Systems* 35.18 (2022), e5348. doi: 10.1002/dac.5348 (cit. on pp. 3, 6, 8, 9, 11, 36, 73, 74, 83, 106).
- [233] E. T. d. Silva, J. Macedo, and A. Costa. "CMAF: Context and Mobility-Aware Forwarding Model for V-NDN". In: *Electronics* 13.12 (2024). issn: 2079-9292. doi: 10.3390/electronics13122394 (cit. on p. 11).

- [234] E. T. d. Silva, J. M. H. d. Macedo, and A. L. D. Costa. "NDN Content Store and Caching Policies: Performance Evaluation". In: *Computers* 11.3 (2022). issn: 2073-431X. doi: 10.3390/computers11030037 (cit. on pp. 8–10, 106).
- [235] V. da Silva, M. Campista, and L. Costa. "TraC: A Trajectory-aware Content distribution strategy for vehicular networks". In: *Vehicular Communications* 5 (2016), pp. 18–34. doi: 10.1016/j.vehcom.2016.09.005 (cit. on pp. 43, 52, 53, 65, 67).
- [236] R. Soua et al. "SDN Coordination for CCN and FC Content Dissemination in VANETs". In: *Ad Hoc Networks*. Ed. by Y. Zhou and T. Kunz. Cham: Springer International Publishing, 2017, pp. 221–233. isbn: 978-3-319-51204-4 (cit. on p. 72).
- [237] A. M. de Sousa, F. R. Araújo, and L. N. Sampaio. "A Link-Stability-Based Interest-Forwarding Strategy For Vehicular Named Data Networks". In: *IEEE Internet Computing* 22.3 (2018), pp. 16–26. doi: 10.1109/MIC.2018.032501512 (cit. on pp. 77, 78, 80).
- [238] Z. Su, Y. Hui, and Q. Yang. "The Next Generation Vehicular Networks: A Content-Centric Framework". In: *IEEE Wireless Communications* 24.1 (2017), pp. 60–66. doi: 10.1109/MWC.2017.1600195WC (cit. on p. 38).
- [239] P. Talebifard et al. "Information-centric networking for VANETs". In: *Vehicular ad hoc Networks: Standards, Solutions, and Research*. Springer International Publishing, 2015-01, pp. 503–524. isbn: 978-3-319-15497-8. doi: 10.1007/978-3-319-15497-8_17 (cit. on pp. 19, 25, 26, 73, 96, 101).
- [240] A. Tariq, R. A. Rehman, and B. Kim. "Forwarding Strategies in NDN-Based Wireless Networks: A Survey". In: *IEEE Communications Surveys Tutorials* 22.1 (2020), pp. 68–95. doi: 10.1109/COMST.2019.2935795 (cit. on pp. 38, 39, 73).
- [241] M. Tavares, O. Aponte, and P. Mendes. "Named-Data Emergency Network Services". In: *Proceedings of the 16th Annual International Conference on Mobile Systems, Applications, and Services*. MobiSys '18. Munich, Germany: Association for Computing Machinery, 2018, p. 516. isbn: 9781450357203. doi: 10.1145/3210240.3210809 (cit. on p. 90).
- [242] R. Tomar, M. Prateek, and H. Sastry. "Vehicular Adhoc Network (VANET)- An introduction". In: 9 (2016-01), pp. 8883–8888 (cit. on p. 17).
- [243] M. Tortelli et al. "ICN software tools: Survey and cross-comparison". In: *Simulation Modelling Practice and Theory* 63 (2016), pp. 23–46. issn: 1569-190X. doi: 10.1016/j.simpat.2016.01.015 (cit. on p. 106).
- [244] S. Totani. "Development and current status of CACS (comprehensive automobile traffic control system)". In: *30th IEEE Vehicular Technology Conference*. Vol. 30. 1980-09, pp. 336–341. doi: 10.1109/VTC.1980.1622831 (cit. on p. 1).

-
- [245] D. Tranfield, D. Denyer, and P. Smart. "Towards a Methodology for Developing Evidence-Informed Management Knowledge by Means of Systematic Review". In: *British Journal of Management* 14 (2003-09), pp. 207–222. doi: 10.1111/1467-8551.00375 (cit. on p. 40).
 - [246] C. Tripp-Barba et al. "Survey on Routing Protocols for Vehicular Ad Hoc Networks Based on Multimetrics". In: *Electronics* 8.10 (2019). issn: 2079-9292. doi: 10.3390/electronics8101177 (cit. on p. 5).
 - [247] D. Trossen et al. *From Design for Tussle to Tussle Networking: PSIRP Vision and Use Cases (TR08-0001)*. Tech. rep. International Telecommunication Union (ITU), 2008. url: http://www.psirp.org/files/Deliverables/PSIRP-TR08-0001_Vision.pdf (cit. on p. 19).
 - [248] I. Ud Din et al. "Left-Right-Front Caching Strategy for Vehicular Networks in ICN-Based Internet of Things". In: *IEEE Access* 9 (2021), pp. 595–605. issn: 2169-3536. doi: 10.1109/ACCESS.2020.3046887 (cit. on pp. 45, 52, 54, 69).
 - [249] R. Ullah, M. Atif Ur Rehman, and B.-S. Kim. "Hierarchical Name-Based Mechanism for Push-Data Broadcast Control in Information-Centric Multihop Wireless Networks". In: *Sensors* 19 (2019-07). doi: 10.3390/s19143034 (cit. on p. 89).
 - [250] D. Van et al. "Efficient caching strategy in content-centric networking for vehicular ad-hoc network applications". In: *IET Intelligent Transport Systems* 12.7 (2018), pp. 703–711 (cit. on pp. 44, 52, 55, 67).
 - [251] M. Varvello et al. "On the design of content-centric MANETs". In: *2011 Eighth International Conference on Wireless On-Demand Network Systems and Services*. 2011-01, pp. 1–8. doi: 10.1109/WONS.2011.5720195 (cit. on pp. 2, 6, 26, 73, 96).
 - [252] A. Vasilakos et al. "Information centric network: Research challenges and opportunities". In: *Journal of Network and Computer Applications* 52 (2015-06). doi: 10.1016/j.jnca.2015.02.001 (cit. on pp. 19, 36).
 - [253] L. A. Villas et al. "DRIVE: An efficient and robust data dissemination protocol for highway and urban vehicular ad hoc networks". In: *Comput. Networks* 75 (2014), pp. 381–394. url: <https://api.semanticscholar.org/CorpusID:9445440> (cit. on p. 73).
 - [254] A. Wang et al. "NDNVIC: Named Data Networking for Vehicle Infrastructure Cooperation". In: *IEEE Access* 7 (2019), pp. 62231–62239. doi: 10.1109/ACCESS.2019.2913656 (cit. on pp. 44, 47, 50, 63).
 - [255] C. Wang et al. "Popularity Incentive Caching for Vehicular Named Data Networking". In: *IEEE Transactions on Intelligent Transportation Systems* (2020), pp. 1–14. doi: 10.1109/TITS.2020.3038924 (cit. on pp. 45, 52, 68).

- [256] F. Wang and R. Hou. "Delayed Broadcast-Based Data Forwarding Strategy for Vehicular Named Data Networks". In: *Workshop on Computing, Networking and Communications (CNC)* (2024) (cit. on pp. 76, 77, 80).
- [257] G. Wang et al. "c-Through: part-time optics in data centers". In: *SIGCOMM Comput. Commun. Rev.* 41.4 (2010-08), pp. –. issn: 0146-4833 (cit. on p. 98).
- [258] J. Wang et al. "A Mobility-Predict-based Forwarding Strategy in Vehicular Named Data Networks". In: *GLOBECOM 2020 - 2020 IEEE Global Communications Conference*. 2020, pp. 01–06. doi: 10.1109/GLOBECOM42002.2020.9322164 (cit. on pp. 73, 74, 80).
- [259] J. Wang et al. "Towards Predictive Forwarding Strategy in Vehicular Named Data Networking". In: *IEEE Transactions on Vehicular Technology* 72.3 (2023), pp. 3751–3763. doi: 10.1109/TVT.2022.3209539 (cit. on pp. 73, 75, 80).
- [260] L. Wang et al. "Data naming in Vehicle-to-Vehicle communications". In: *2012 Proceedings IEEE INFOCOM Workshops*. 2012-03, pp. 328–333. doi: 10.1109/INFCOMW.2012.6193515 (cit. on p. 2).
- [261] L. Wang et al. "Investigating the Synergy between Routing and Forwarding Strategy in NDN Networks". In: *Proceedings of the 10th ACM Conference on Information-Centric Networking*. ACM ICN '23. Reykjavik, Iceland: Association for Computing Machinery, 2023, pp. 67–77. isbn: 9798400704031. doi: 10.1145/3623565.3623714 (cit. on pp. 83, 97, 139).
- [262] L. Wang et al. "Rapid Traffic Information Dissemination Using Named Data". In: *Proceedings of the 1st ACM Workshop on Emerging Name-Oriented Mobile Networking Design - Architecture, Algorithms, and Applications*. NoM '12. Hilton Head, South Carolina, USA: Association for Computing Machinery, 2012, pp. 7–12. isbn: 9781450312912. doi: 10.1145/2248361.2248365 (cit. on pp. 76, 78, 80).
- [263] X. Wang et al. "A new content-centric routing protocol for Vehicular Ad Hoc Networks". In: *2016 22nd Asia-Pacific Conference on Communications (APCC)*. 2016, pp. 552–558. doi: 10.1109/APCC.2016.7581473 (cit. on pp. 43, 47, 48, 62).
- [264] *Web of Science*. 2020-03. url: <https://www.webofknowledge.com/> (cit. on p. 41).
- [265] Y. Wei et al. "A novel dynamic adaptive video streaming solution in content-centric mobile network". In: 2016. doi: 10.1109/PIMRC.2016.7794977 (cit. on pp. 43, 52, 53, 65–67).
- [266] Z. Wei et al. "Data Forwarding and Caching Strategy for RSU Aided V-NDN". In: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* 11604 LNCS (2019), pp. 605–612. doi: 10.1007/978-3-030-23597-0\$__\$50\$ (cit. on pp. 44, 52, 54, 68).
- [267] G. Welch and G. Bishop. *An Introduction to the Kalman Filter*. Tech. rep. USA, 1995 (cit. on pp. 28, 29).

- [268] G. Welch and G. Bishop. "An Introduction to the Kalman Filter". In: *Proc. Siggraph Course 8* (2006-01) (cit. on p. 31).
- [269] N. Wiki. *ccnd 0.7.2 forwarding strategy*. 2022-12. url: <https://redmine.named-data.net/projects/nfd/wiki/CcndStrategy> (cit. on p. 107).
- [270] B. Wessingh et al. *Information-Centric Networking (ICN): Content-Centric Networking (CCNx) and Named Data Networking (NDN) Terminology*. RFC 8793. RFC Editor, 2020-06 (cit. on p. 19).
- [271] R. Wu et al. "Location prediction on trajectory data: A review". In: *Big Data Mining and Analytics* 1.2 (2018), pp. 108–127. doi: 10.26599/BDMA.2018.9020010 (cit. on p. 102).
- [272] C. Xu et al. "GrlMS: Green Information-Centric Multimedia Streaming Framework in Vehicular Ad Hoc Networks". In: *IEEE Transactions on Circuits and Systems for Video Technology* 28.2 (2016), pp. 483–498. doi: 10.1109/TCSVT.2016.2607764 (cit. on pp. 43, 52, 56, 65–67).
- [273] C. Xu et al. "Video Streaming in Content-Centric Mobile Networks: Challenges and Solutions". In: *IEEE Wireless Communications* 24.5 (2017), pp. 157–165. doi: 10.1109/MWC.2017.1600219WC (cit. on p. 47).
- [274] W. Xu et al. "NIHR: Name/ID Hybrid Routing in Information-centric VANET". In: *2020 IEEE Wireless Communications and Networking Conference (WCNC)*. 2020, pp. 1–7 (cit. on pp. 45, 47, 50, 63).
- [275] Y. Xu et al. "COMPASS: Directing Named Data Transmission in VANETs by Dynamic Directional Interfaces". In: *IEEE Access* 8 (2020), pp. 8418–8435. doi: 10.1109/ACCESS.2019.2963547 (cit. on pp. 44, 47, 50, 63).
- [276] G. Xylomenos et al. "A Survey of Information-Centric Networking Research". In: *IEEE Communications Surveys Tutorials* 16.2 (2014), pp. 1024–1049. doi: 10.1109/SURV.2013.070813.00063 (cit. on pp. 2, 20).
- [277] Z. Yan, S. Zeadally, and Y. Park. "A Novel Vehicular Information Network Architecture Based on Named Data Networking (NDN)". In: *IEEE Internet of Things Journal* 1.6 (2014-12), pp. 525–532. issn: 2372-2541. doi: 10.1109/JIOT.2014.2354294 (cit. on pp. 2, 38, 43, 46, 47, 60, 62, 83).
- [278] N. Yang, K. Chen, and Y. Liu. "Towards Efficient NDN Framework for Connected Vehicle Applications". In: *IEEE Access* 8 (2020), pp. 60850–60866. doi: 10.1109/ACCESS.2020.2981928 (cit. on pp. 45, 47, 50, 63).
- [279] L. Yao et al. "A Cooperative Caching Scheme Based on Mobility Prediction in Vehicular Content Centric Networks". In: *IEEE Transactions on Vehicular Technology* 67.6 (2017-06), pp. 5435–5444. issn: 1939-9359. doi: 10.1109/TVT.2017.2784562 (cit. on p. 102).

- [280] L. Yao et al. "A Cooperative Caching Scheme Based on Mobility Prediction in Vehicular Content Centric Networks". In: *IEEE Transactions on Vehicular Technology* 67.6 (2018-06), pp. 5435–5444. issn: 1939-9359. doi: 10.1109/TVT.2017.2784562 (cit. on pp. 44, 52, 65–67).
- [281] L. Yao et al. "A Pre-caching Mechanism of Video Stream Based on Hidden Markov Model in Vehicular Content Centric Network". In: *2020 3rd International Conference on Hot Information-Centric Networking (HotICN)*. 2020-12, pp. 53–58. doi: 10.1109/HotICN50779.2020.9350830 (cit. on pp. 45, 52, 54, 65, 69).
- [282] L. Yao et al. "Cooperative Caching in Vehicular Content Centric Network based on Social Attributes and Mobility". In: *IEEE Transactions on Mobile Computing* (2019), pp. 1–1 (cit. on pp. 44, 52, 53, 65, 66, 68).
- [283] L. Yao et al. "Detection and Defense of Cache Pollution Attacks Using Clustering in Named Data Networks". In: *IEEE Transactions on Dependable and Secure Computing* 17.6 (2020), pp. 1310–1321. doi: 10.1109/TDSC.2018.2876257 (cit. on p. 57).
- [284] L. Yao et al. "Exploiting Non-Cooperative Game Against Cache Pollution Attack in Vehicular Content Centric Network". In: *IEEE Transactions on Dependable and Secure Computing* (2021), pp. 1–1. doi: 10.1109/TDSC.2021.3109046 (cit. on pp. 45, 57, 58, 70).
- [285] L. Yao et al. "Popularity Prediction Caching Using Hidden Markov Model for Vehicular Content Centric Networks". In: *2019 20th IEEE International Conference on Mobile Data Management (MDM)*. 2019-06, pp. 533–538. doi: 10.1109/MDM.2019.00115 (cit. on pp. 44, 52, 56, 64, 66).
- [286] M. A. Yaqub et al. "Interest Forwarding in Vehicular Information Centric Networks: A Survey". In: *Proceedings of the 31st Annual ACM Symposium on Applied Computing*. SAC '16. Pisa, Italy: Association for Computing Machinery, 2016, pp. 724–729. isbn: 9781450337397. doi: 10.1145/2851613.2851857 (cit. on pp. 38, 39).
- [287] B. Yi, X. Wang, and M. Huang. "Content delivery enhancement in Vehicular Social Network with better routing and caching mechanism". In: *Journal of Network and Computer Applications* 177 (2021). doi: 10.1016/j.jnca.2020.102952 (cit. on pp. 45, 49, 52, 55, 63, 69).
- [288] C. Yi et al. "On the Role of Routing in Named Data Networking". In: *Proceedings of the 1st ACM Conference on Information-Centric Networking*. ACM-ICN '14. Paris, France: Association for Computing Machinery, 2014, pp. 27–36. isbn: 9781450332064. doi: 10.1145/2660129.2660140 (cit. on pp. 23, 24, 83, 96).
- [289] K. Yu et al. "Information-Centric Networking: Research and Standardization Status". English. In: *IEEE Access* 7 (2019), pp. 126164–126176. issn: 2169-3536. doi: 10.1109/ACCESS.2019.2938586 (cit. on pp. 2, 20).

- [290] X. Yu et al. "A distance-based interest forwarding protocol for vehicular information-centric networks". In: *2017 IEEE 28th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC)*. 2017, pp. 1–5. doi: 10.1109/PIMRC.2017.8292229 (cit. on pp. 76, 77, 80).
- [291] Y.-T. Yu et al. "Interest propagation in named data manets". In: *2013 International Conference on Computing, Networking and Communications (ICNC)*. 2013, pp. 1118–1122. doi: 10.1109/ICCNC.2013.6504249 (cit. on pp. 78, 80).
- [292] Y.-T. Yu et al. "Scalable opportunistic VANET Content Routing with encounter information". In: *2013 21st IEEE International Conference on Network Protocols (ICNP)*. 2013-10, pp. 1–6. doi: 10.1109/ICNP.2013.6733679 (cit. on pp. 43, 47, 49, 61, 97).
- [293] Y. Yu et al. "Scalable VANET content routing using hierarchical bloom filters". In: *2013 9th International Wireless Communications and Mobile Computing Conference (IWCMC)*. 2013, pp. 1629–1634 (cit. on pp. 43, 46–48, 59–61, 97, 98).
- [294] W. U. I. Zafar et al. "Context-Aware Naming and Forwarding in NDN-Based VANETs". In: *Sensors* 21.14 (2021). issn: 1424-8220. doi: 10.3390/s21144629 (cit. on pp. 76, 77, 80).
- [295] W. U. I. Zafar et al. "Decentralized Receiver-based Link Stability-aware Forwarding Scheme for NDN-based VANETs". In: *Computer Networks* 236 (2023), p. 109996. issn: 1389-1286. doi: 10.1016/j.comnet.2023.109996 (cit. on pp. 2, 76, 77, 80).
- [296] K. Zaidi and M. Rajarajan. "Vehicular internet: Security and privacy challenges and opportunities". In: 7 (2015-01), pp. 257–275. doi: 10.3390/fi7030257 (cit. on p. 17).
- [297] G. Zhang, Y. Li, and T. Lin. "Caching in Information Centric Networking: A Survey". In: *Comput. Netw.* 57.16 (2013-11), pp. 3128–3141. issn: 1389-1286. doi: 10.1016/j.comnet.2013.07.007 (cit. on p. 38).
- [298] H. Zhang and L. Dai. "Mobility Prediction: A Survey on State-of-the-Art Schemes and Future Applications". In: *IEEE Access* 7 (2019), pp. 802–822 (cit. on pp. 9, 26–28, 57, 101).
- [299] H. Zhang et al. "Deep Learning Based Traffic and Mobility Prediction". In: *Machine Learning for Future Wireless Communications*. John Wiley and Sons, Ltd, 2020. Chap. 7, pp. 119–136. isbn: 9781119562306. doi: 10.1002/9781119562306.ch7 (cit. on pp. 28, 29, 45, 47, 49, 63).
- [300] L. Zhang, J. Burke, and V. Jacobson. *FIA-NP: Collaborative Research: Named Data Networking Next Phase (NDN-NP)*. Tech. rep. NDN, 2014 (cit. on p. 21).
- [301] L. Zhang et al. "Named data networking." In: *Computer Communication Review* 44.3 (2014), pp. 66–73 (cit. on pp. 2, 19, 36, 39).
- [302] L. Zhang et al. *Named Data Networking (NDN) Project*. Technical Report NDN-001. NDN, 2010 (cit. on pp. 19, 20, 22, 23, 37).

- [303] M. Zhang, H. Luo, and H. Zhang. "A Survey of Caching Mechanisms in Information-Centric Networking". In: *IEEE Communications Surveys Tutorials* 17.3 (2015), pp. 1473–1499. doi: 10.1109/COMST.2015.2420097 (cit. on p. 39).
- [304] X. Zhang et al. "Detection of malicious nodes in NDN VANET for Interest Packet Flooding Broadcast Diffusion Attack". In: *2017 11th IEEE International Conference on Anti-counterfeiting, Security, and Identification (ASID)*. 2017-10, pp. 114–118. doi: 10.1109/ICASID.2017.8285755 (cit. on pp. 44, 57, 59).
- [305] Y. Zhang et al. "A survey of mobility support in Named Data Networking". In: *2016 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*. 2016, pp. 83–88. doi: 10.1109/INFCOMW.2016.7562050 (cit. on pp. 38, 39).
- [306] Z. Zhang et al. "Smart Proactive Caching: Empower the Video Delivery for Autonomous Vehicles in ICN-Based Networks". In: *IEEE Transactions on Vehicular Technology* 69.7 (2020), pp. 7955–7965. doi: 10.1109/TVT.2020.2994181 (cit. on pp. 45, 52, 53, 65, 68).
- [307] W. Zhao et al. "An Efficient Cache Strategy in Information Centric Networking Vehicle-to-Vehicle Scenario". In: *IEEE Access* 5 (2017), pp. 12657–12667. issn: 2169-3536. doi: 10.1109/ACCESS.2017.2714191 (cit. on pp. 44, 52, 54, 56, 65, 67).
- [308] Z. Zhao et al. "LSTM network: a deep learning approach for short-term traffic forecast". In: *IET Intelligent Transport Systems* 11.2 (2017), pp. 68–75. issn: 1751-9578. doi: 10.1049/iet-its.2016.0208 (cit. on p. 53).
- [309] J. Zhou et al. "Cache Pollution Prevention Mechanism Based on Cache Partition in V-NDN". In: *2020 IEEE/CIC International Conference on Communications in China (ICCC)*. 2020-08, pp. 330–335. doi: 10.1109/ICCC49849.2020.9238838 (cit. on pp. 45, 57, 58, 70).
- [310] X. Zhou. "Study on ring signature and its application". In: *2009 Chinese Control and Decision Conference*. 2009, pp. 1388–1393. doi: 10.1109/CCDC.2009.5191586 (cit. on p. 59).
- [311] Z. Zhu et al. "Recent advances in connected vehicles via information-centric networking". In: *IET International Conference on Intelligent and Connected Vehicles (ICV 2016)*. 2016, pp. 1–8. doi: 10.1049/cp.2016.1162 (cit. on p. 39).