An efficient information-centric mechanism for the next generation vehicular networks

Francisco de Carvalho Natário

Thesis to obtain the Master of Science Degree in

Electrical and Computer Engineering

Supervisors: Prof. Naercio David Pedro Magaia
Prof. Paulo Rogério Barreiros D'Almeida Pereira

Examination Committee

Chairperson: Prof. João Luís Da Costa Campos Gonçalves Sobrinho
Supervisor: Paulo Rogério Barreiros D'Almeida Pereira
Members of the Committee: Prof. Miguel Nuno Dias Alves Pupo Correia

November 2022

I declare that this document is an original work of my own authorship and that it fulfils all the requirements of the Code of Conduct and Good Practices of the Universidade de Lisboa.

Acknowledgments

Em primeiro lugar gostaria de agradecer ao Professor Paulo Rogério Barreiros D'Almeida Pereira e ao Professor Naercio David Pedro Magaia por me terem conduzido ao longo deste projeto. Obrigado por investirem o vosso tempo e esforço, e me acompanharem tão proximamente. Sem vocês não teria sido possível.

Ao Prof Naercio Magaia por apoiar o projeto nos seus primeiros e longos passos e por fim a encarrilá-lo na direção e caminhos certos. Ao Professor Paulo Pereira pelo constante apoio, dedicação e resiliência a este projeto desde do primeiro ao último dia.

Agradeço também aos meus amigos que me apoiaram neste projeto e que tantas palavras de encorajamento deram e que cuja presença foi essencial em tempos difíceis.

Um especial agradecimento à Inês que esteve ao meu lado, muitas vezes literalmente, todos os dias por altos e baixos com o seu apoio, carinho, atenção e amor, sem ela este projeto não teria sido possível.

Por último, um especial agradecimento à minha família que esteve presente em cada etapa deste percurso com carinho e amor.

Abstract

As a crucial component of Intelligent Transportation Systems (ITS), the Internet of Vehicles (IoV) plays an essential role in intelligent cities. However, current Internet protocol solutions cannot guarantee efficient data delivery in highly dynamic mobile environments. Therefore, Named Data Networking (NDN), a leading architecture of Information-Centric Networking (ICN), is introduced into IoV to focus on transmitting and retrieving content, regardless of its physical location. The NDN architecture in-network caching and IP address decentralization characteristics are crucial to surpass IoV environment challenges. While the nature of the NDN architecture essentially supports consumer mobility, producer mobility faces complex challenges. Producer mobility relies on up-to-date Consumer routing tables to minimize Interest packet delivery loss and Consumer delay in IoV scenarios.

In this context, this work proposes an anchor-less signalling approach to the NDN data plane, aiming to mitigate the problems posed by the mobility of content producers in dynamic networks such as IoV. The proposed mechanism ensures a reduced disruption in the data delivery process by using a unique Interest forwarding mechanism without relying on the existing routing protocol. We analyze the performance of the proposed approach against default NDN architecture regarding content consumer delay, consumer satisfaction, and network overhead.

Keywords: Internet of Vehicles, Named Data Networking, Producer, Mobility;

Resumo

Como um componente crucial dos *Intelligent Transportation Systems* (ITS), a *Internet of Vehicles* (IoV) desempenha um papel essencial em cidades inteligentes. No entanto, as soluções atuais baseadas no *Internet Protocol* (IP) não podem garantir a entrega eficiente de dados em ambientes móveis dinâmicos. *Named Data Networking* (NDN), uma arquitetura líder de *Information-Centric Networking* (ICN), é escolhida para que IoV beneficie do foco que NDN têm na transmissão e recuperação de conteúdo, não importando a sua localização física. A arquitetura NDN utiliza os conteúdos reservados em cache na rede e na descentralização do conteúdo do seu endereço IP para ultrapassar as dificuldades introduzidas pelo ambiente IoV. Embora a mobilidade do consumidor seja suportada inerentemente pela natureza da arquitetura NDN, a mobilidade do produtor continua a enfrentar desafios. A mobilidade do produtor depende de tabelas de roteamento atualizadas para minimizar a perda desnecessária de pacotes de interesse e o seu atraso em cenários IoV.

Neste contexto, este trabalho propõe uma abordagem de sinalização sem âncora para o plano de dados NDN, visando mitigar os problemas colocados pela mobilidade de produtores de conteúdo em redes dinâmicas como IoV. O mecanismo proposto garante uma entrega bem sucedida de dados usando um mecanismo de encaminhamento de interesse exclusivo, sem depender da existência de um protocolo de roteamento. Analisamos o desempenho da abordagem proposta em relação à arquitetura NDN padrão em relação à disponibilidade de conteúdo, satisfação do consumidor e sobrecarga na rede.

Keywords: Internet of Vehicles, Named Data Networking, Produtor, Mobilidade

Contents

List of Tables

List of Figures

Acronyms

API Application Programming Interface.

AU Application Unit.

BGP Border Gateway Protocol.

CAC Courier-Assisted Cooperation.

CCN Content-Centric Network.

CDBMA Control/Data-Based Mobility Approach.

CRoS Controller-based Routing Scheme.

CS Content Store.

CSMA Carrier Sense Multiple Access.

DADT Density-Aware Delay-Tolerant.

DHT Distributed Hash Table.

DoS Denial of Service.

DSRC Dedicated Short-Range Communications.

FCI Flat Content Identifier.

FIB Forwarding Information Base.

GPS Global Positioning System.

HRP Hierarchical Routable Prefix.

IBMA Indirection Based Mobility Approach.

ICN Information-Centric Network.

IEEE Institute of Electrical and Electronics Engineers.

IN Interest Notification.

IoT Internet of Things.

IoV Internet of Vehicles.

IP Internet Protocol.

ISD Interest Satisfaction Delay.

ISP Internet Service Provider.

IST Instituto Superior Técnico.

ITS Intelligent Transportation Systems.

IU Interest Update.

LBMA Location-Based Mobility Approach.

LRU Least Recently Used.

LTE Long Term Evolution.

MAC Media Access Control.

MANET Mobile Ad-Hoc Network.

MBMA Mapping Based Mobility Approach.

NCB Notification Controlled Broadcast.

NDN Named Data Networking.

NDN-Q NDN Query mechanism.

NDO Named Data Object.

NETInf Network of Information.

NFD Named Data Networking Forwarder Daemon.

NLSR Named-data Link State Routing.

NN Neighbour Node.

NS-3 Network Simulator 3.

OBU Onboard Unit.

OSPFN Open Shortest Path First.

PAC Partner Assisted Cooperation.

PALS Primary Attribute Labels.

PD Producer Discovery.

PDA Personal Digital Assistant.

PDCN Producer Data Controlled Notification.

PDCNwDM Producer Data Controlled Notification with Degree Method.

PDV Popularity-Density Value.

PIT	Pending Interest Table.
PSIRP	Publish-Subscribe Internet Routing Paradigm.
PURSUIT	Publish-Subscribe Internet Technology.
QoE	Quality of Experience.
QoS	Quality of Service.
RFID	Radio Frequency Identification.
RSSI	Received Signal Strength Indicator.
RSU	Roadside Unit.
SAIL	Scalable and Adaptive Internet Solutions.
SMM	Scalable Mobility Management.
SUMO	Simulation of Urban Mobility.
URL	Uniform Resource Locator.
V2H	Vehicle to Home.
V2I	Vehicle to Infrastructure.
V2R	Vehicle-to-roadside.
V2S	Vehicle to Sensor.
V2V	Vehicle-to-vehicle.
V2X	Vehicle-to-everything.
VANET	Vehicular Ad-hoc Network.
VNDN	Vehicular Named Data Network.
wi-fi	Wireless Fidelity.
WiMAX	Worldwide Interoperability for Microwave Access.
WMN	Wireless Mesh Network.
WSN	Wireless Sensor Network.

Chapter 1

Introduction

Mobility has become an essential requirement for almost any communication network. The need for a mobility management paradigm to apply within IP networks resulted in a complex set of mechanisms implemented via a dedicated control infrastructure. The complexity and lack of flexibility of such approaches (e.g., Mobile IP [8]) promoted the development of promising Information-centric Network (ICN) [9] architectures as a paradigm shift from the traditional IP-based communication.

ICN approaches were developed to answer the ever-increasing urge from users to access information irrespective of its physical location while helping to address a series of limitations in the current Internet architecture, e.g. mobility management in vehicular networks. ICN characteristics such as named content retrieval, innate multicast support, and in-network data caching are some of the advantages that surpass the challenging demands of vehicular networks, and their evolution [4]. Such challenges encompass mobility management and new requirements such as seamless, secure, robust, scalable information exchange among cars, humans, and roadside infrastructures. Furthermore, the Internet of Vehicles (IoV) prioritizes information (e.g., traffic and congestion) rather than a node identity. It requires information to be propagated in environments with highly dynamic vehicle movement while minimizing latency due to an increasing number of connected vehicles.

In ICN, the content-based approach means that the content is requested by its name to retrieve from the network, decoupling content from its sources. The way for a user to get the content it desires is by its name, i.e. its identifier, without referring to the node's IP address storing the content. We choose the Named Data Networking (NDN) [10] framework among other ICN architectures as significant attention from the research community has been focused on its development and maturity.

The NDN communication is based on three elements: the Consumer nodes which aims to get data that may be provided by another element in the network; the Producer nodes which provide the requested content; the router nodes that may cache transit data aiming to support future requests.

A Consumer sends out an Interest packet that carries a name to identify the desired information. The Consumer drives communication in NDN. A router remembers the interface from which the request comes in and then forwards the Interest packet by looking up the name in its Forwarding Information Base (FIB), populated by a name-based routing protocol. Once the Interest reaches a node with the requested Data, the packet is sent back. The Data packet carries the name, the content and a Producer signature key. This Data packet follows in reverse the path taken by the Interest to get back to the Consumer: Data packets are forwarded based on the state set up by Interests in the Pending Interest Table (PIT) at each router.

1.1 Motivation

The NDN architecture provides natural support for Consumer mobility (e.g., the Consumer can keep requesting information until it receives the desired content while changing domains). However, the NDN architecture does not support Producer mobility which faces many challenges, further emphasized in a highly dynamic environment [11] [12].

While the presence of in-network caches may help to smooth the effect of the Producer mobility, it can still cause unnecessary Interest packet loss for nodes without an updated forwarding information base, causing congestion that affects data integrity. Without a dynamic routing mechanism to ensure Interest broadcast control, consumers' continuous requests for content can lead to broadcast storm problems.

Furthermore, losing Interest and increasing network congestion due to the lack of forwarding information during producer movement can lead to profound implications for the quality of service of time-sensitive applications.

1.2 Objectives

To tackle such problems, we use the existing NDN Interest/Data packet structures to trace Producer movements and build a reverse-forwarding path dynamically. We develop a proactive signalling mechanism to update the FIB tables at intermediate nodes to keep the Producer content reachable after its movement, even in the absence of a routing protocol.

We propose PDCN (Producer Data Controlled Notification), which intends to solve the Producer mobility problem through the data plane.

We aim to solve the Producer mobility problem while assuring high packet delivery success, Interest broadcast minimization and low packet delivery delay in a highly dynamic network.

1.3 Outline

The Thesis is organized in the following order:

Chapter 2 addresses essential concepts to understand the Thesis subject. The chapter approaches the concept of vehicular networks and their evolution from ITS to IoV. It introduces the ICN model and its characteristics, particularly its NDN instantiation. The chapter details the NDN simulation tool on which the proposal was implemented and tested. Finally, the chapter analyses different Producer mobility support schemes and related mobility problems.

Chapter 3 introduces the implemented architecture, the PDCN functionality and the performance metrics chosen to evaluate PDCN.

Chapter 4 presents the simulation results for evaluation of the developed architecture, the relationship between mobility, caching and content availability, and the architecture performance analysis.

Chapter 5 concludes the Thesis and shows future work suggestions.

Chapter 2

State of the Art

2.1 Vehicular Networks

2.1.1 Intelligent Transportation System

The number of vehicles has increased over the last five decades [13] [14]. This growth increases accidents, traffic congestion, transportation delays, and more significant vehicle pollution emissions. Dimitrakopoulos et al. [15] showed that more and more vehicles are responsible for the plethora of traffic jams and accidents. In large cities, exist a high number of services, and human beings, which in turn requires better services to satisfy the city and the people's necessities [13]. Intelligent Transportation System (ITS) combines developed technology and information systems, sensors, actuators and mathematical methods with the transportation system to manage traffic and vehicles to guarantee road safety [15]. For example, in an accident, vehicles can warn other vehicles to avoid further congestion in the area, and it can create different routes to the vehicle's destination, reducing the traffic jams and ensuring users' and vehicles' safety [16]. For this reason, ITS has been developed over three decades to solve the increasing social, and economic impacts in the transportation system [17].

Moloisane et al. [18] showed that the fixed time traffic light control (the current method used to regulate traffic) is not efficient. The traffic control does not change its behaviour in the face of evolving traffic flow and can not control possible congestion resulting from the daily flow of traffic. Examples like this prove the need and importance of intelligent and dynamic systems that allow communication between vehicles, road infrastructures and pedestrians. The integration of these technologies can significantly alter and improve the traditional paradigm of integration between vehicles, infrastructures, and people [18].

The current problems in the transportation system are: (i) Human error,which is the cause of most road accidents [13] [17]. (ii) Investment difficulties, in new infrastructures or old ones such as highways, due to the lack of resources [13] [2]. The lack of resources and the permanent impact of human error created the need to minimize these impacts by developing a strategy. Zhang et al. [17] propose some techniques that are not cost-effective such as imposing a car usage restriction to avoid air pollution and road congestion. On the other hand, Zhang et al. [17] also proposed a way to improve the current paradigm by optimizing the transportation system. The optimization proposed is only possible with the help of a plethora of instruments and a wide range of intelligent devices often equipped with embedded processors and wireless communication technologies. These smart devices can generate, process, and transmit data essential for improving road safety and traffic efficiency, where VANET (Vehicle ad hoc Network) plays a significant role.

2.1.2 Vehicular Ad Hoc Network

Vehicular Ad Hoc Network (VANET) provides wireless communication between vehicles or between vehicles and Roadside Units (RSUs) [19]. This communication is possible with a dedicated short-range communication (DSRC) [20].is based on IEEE 802.11a physical layer and IEEE 802.11 MAC layer to support ITS applications.

RSUs, which are connected to the backbone internet, are devices dedicated to short-range communication that must be in a fixed place to facilitate communication. For this reason, RSUs are evenly or intermittently spaced depending on the situation, which will create different communication scenarios. RSUs serve as the internet connectivity provider and the primary source of information to Onboard Units (OBUs) and as an extended range of the ad-hoc network by forwarding the data between OBUs.

Each vehicle has an OBU, one or more Application Units (AU) and a set of sensors. AU is a device responsible for running applications, such as a Personal Digital Assistant (PDA). Each OBU device can use the service provided by an RSU or provide itself services, i.e., hosting an application. Furthermore, OBUs' primary functions are to maintain communication(s) with the other OBUs or RSUs. These functions include mobility (geographical routing), network congestion control, reliable message transfer, data security and IP mobility. All these devices establish a vehicle as a service provider and a user.

VANET communication

VANET communication consists of three different types: intra-vehicle, vehicle to vehicle (V2V), and vehicle to infrastructure (V2I) [19]. Intra-vehicle communication is communication between the OBU and AUs or AU by wireless or cable present in the vehicle.

AU only communicates with the network by the OBU. Additionally, the AU executes one or more applications using the communication capabilities of the OBU.

A V2V and ad hoc network allows the exchange of information between cars in a distributed manner and with decentralised coordination. V2V communication is defined as dedicated and direct communication between two vehicles, conducting a wireless connection, i.e. forming a single hop. In an indirect connection, a car can communicate with another by forwarding information through other vehicles (multi-hopping) until it reaches the destination.

V2I communication is defined as the direct communication between a vehicle and an RSU. V2I increases the range of communication between cars by forwarding data from different nodes. On top of that, RSUs can process special applications for the OBUs, sending and receiving service data. Furthermore, an OBU can choose to off-load its tasks to a mobile-edge computing server existing in the RSU [21].

VANET Evolution to IoV

The characteristics present in the connection between the vehicular network and its surroundings are both present in VANET and IoV. Nevertheless, VANET is not enough to represent all the properties and intricacies of the Internet of Vehicles, and that is why VANET is evolving into the Internet of Vehicles [2].

According to Contreras-Castillo et al. [2], there are three main reasons for VANET not being enough to satisfy the current needs of ITS. The main reasons are:

- The lack of processing capacity for handling vast amounts of data
- the incapacity to analyse, process and evaluate the global information that is collected from the different vehicles in the network

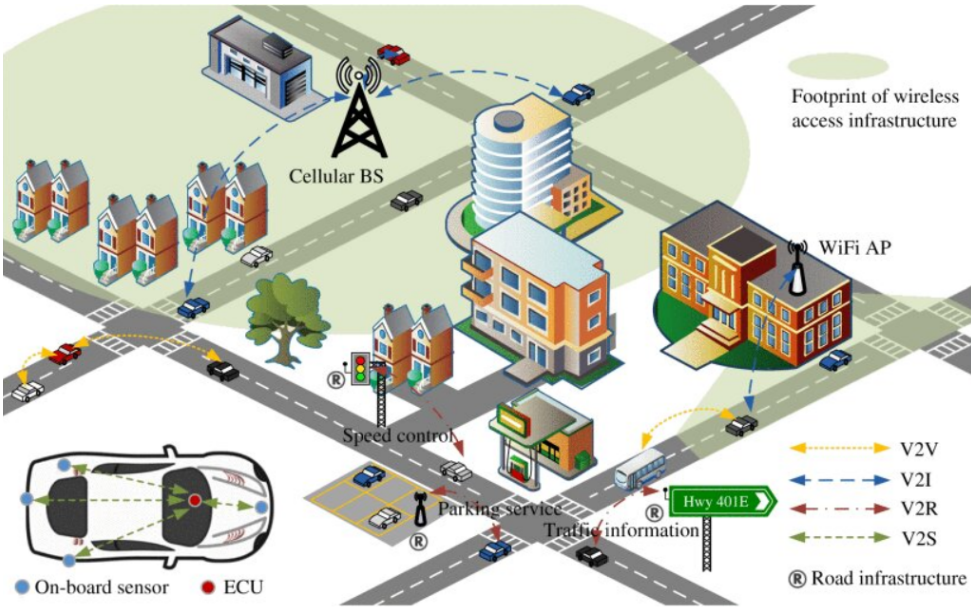


Figure 2.1: IoV model [1]

- The lack of mobility affected by the number of connected vehicles and different environmental factors such as traffic jams, tall buildings and deviant driver behaviour.

Contreras-Castillo et al. show that IoV integrates two technological visions: the vehicle's networking and intelligence. IoV combines the environment (people, cars and things) and creates a complex and interconnected network with different services such as traffic management, road safety and infotainment. This interconnected network is much bigger and more complex than the network created by the wireless connection of cars that is VANET.

2.1.3 Internet of Vehicles

Internet of Vehicles (IoV) is focused on the exchange of information between the vehicles and the devices in the environment through different communication media and by an IP-based message infrastructure, [2], [15].

The constant technological evolution in the vehicular network paradigm drove the development of VANETs into the IoV. IoV adds a wider communication variety to the existing VANET and expands its network, which includes intra-vehicular communication systems such as Vehicle to Sensor (V2S), and more inter-vehicular communication such as Vehicle to Infrastructure (V2I), Vehicle to Human (V2H) and vehicle-to-Internet. Figure 2.1 is an example of the IoV environment, presented by Lu et al. [1]. The IoV also differentiates from VANET in the variety of communication solutions, i.e., vehicles in IoV can communicate through 4G and DSRC [22].

IoV is a vast network where vehicles exchange information with infrastructures, humans, and sensors, among others, to support different functions (such as dynamic information services, intelligent traffic management, intelligent vehicle control and more) [23].

The IoV paradigm, where the dynamic mobile communication systems are capable of sharing, processing, storing and computing information, enabled the ITS evolution [1].

IoV applications and Challenges

Integrating the technologies of the global position system (GPS) technologies, radio frequency identification (RFID), data mining and automatic control, the IoV realizes intelligent traffic management, intelligent dynamic information service, and intelligent vehicle control [24].

IoV uses Vehicular-to-everything (V2X) technology to break non-line-of-sight perception and vehicle information sharing. IoV is expected to improve the intelligence level and autonomous driving ability and satisfy the requirements of many services and applications such as vehicular safety applications, road traffic efficiency and infotainment applications, and Video streaming [24].

Applications in an IoV environment need specific characteristics to function correctly. Some of those characteristics are [25]:

- High scalability network: IoV environments are full of interacting entities and elements, creating a necessity for a large-scale network.
- Dynamic Topology: The vehicle's dynamic movement and velocity create an ever-changing network topology.
- Non-Uniform Network Density: Vehicle networks change with vehicle density and geographical location conditions, e.g. city and highway.
- Complex Communication: The vehicle's speed and relative position interfere with the time for communication. For example, high-speed vehicles require minimal communication delay with unmoving traffic lights.

This complex communication makes IoV prone to some difficult communication challenges [25]:

- Poor network connectivity: Remote locations have network connectivity issues which generate obstacles for the correct function of IoV.
- Lack of standards: Different vehicles have different ways of communicating and connecting, making V2V ineffective. Open standards and integrated systems are essential for smooth information sharing.
- Delay constraints and fault tolerance: IoV applications require low service delay and reliable network communication, i.e., real-time communication, even when communication or network nodes malfunction.
- Security and privacy: Incorrect or fabricated information can cause severe damage to vehicles and can lead to the loss of human lives. Intruders or hackers can hack into the system, leading to accidents. Identification of vehicles, user data, and travelling information are serious privacy violations that can lead to many unknown repercussions.

We intend to tackle delay constraints and fault tolerance requirements with the proposed solution discussed in subsequent sections.

Architecture

Contreras-Castilho et al. [23] propose a seven-layer IoV architecture after analysing different IoV architectures from different researchers. The architectural model presented in figure 2.2 allows a transparent interconnection of all the network components and dissemination of data in an IoV environment. We further analyse each layer to shed some light on their importance and context.

- User interface layer:

  This layer provides direct interaction with the driver. This interaction is made through an interface to inform the driver about something as crucial as a collision risk. The interface manages and coordinates all driver notifications and selects the best way to display them to the driver. For example, when information about a collision occurrence arrives at the vehicle, the interface can display the data using a set of lights on the instrument board.

- Data acquisition layer:

  This layer aims to gather relevant data from various sources. The data comes from sources such as vehicle's internal sensors, Global Positioning System (GPS), other cars, Wireless Sensor Networks (WSNs) and multi-devices (e.g. mobile devices, traffic lights and road signals).

- Data filtering and Pre-processing layer

  IoV devices communicate and generate vast amounts of data, creating irrelevant data dissemination, resource waste, and network congestion. Thus, this layer analyses data and performs some filtering to reduce network traffic, which is essential for communication performance.

- Communication layer

  In the IoV environment, multiple devices communicate with different wireless networking technologies. Different technologies such as LTE, Bluetooth, Worldwide Interoperability for Microwave Access (WiMAX) and Dedicated short-range communications (DSRC) for V2V, V2R and V2I communication. This heterogeneous network with different characteristics creates difficulties in satisfying users' requirements. Thus, the communication layer aims to select the most suitable network to send information. This selection is based on relevant information (e.g. network congestion) and measurements (e.g. QoS level between different networks).

- Control and management layer

  This layer is responsible for managing different network service providers within the IoV environment. This layer applies other policies, such as packet inspection or traffic management, to control and collect the information generated by all the devices within the IoV environment.

- Processing layer

  The processing layer is responsible for processing, analysing and storing all the information received from the other layers. This layer uses various types of cloud-edge computing infrastructures, locally and remotely, to process such amounts of data. Different applications can apply the resulting data to improve themselves.

- Security layer

  This layer is applied throughout the IoV architecture and is crucial to every layer. This layer is responsible for using security measures such as data authentication, integrity, access control and confidentiality that are necessary to the architecture functioning.

2.2 Information-Centric Network

In the current internet context, an established connection between two nodes is the basis of a communication paradigm. This context emerged from a time when scarce resources were transmitted through a long, well defined and secure path. Nevertheless, new content providers and consumers emerged as

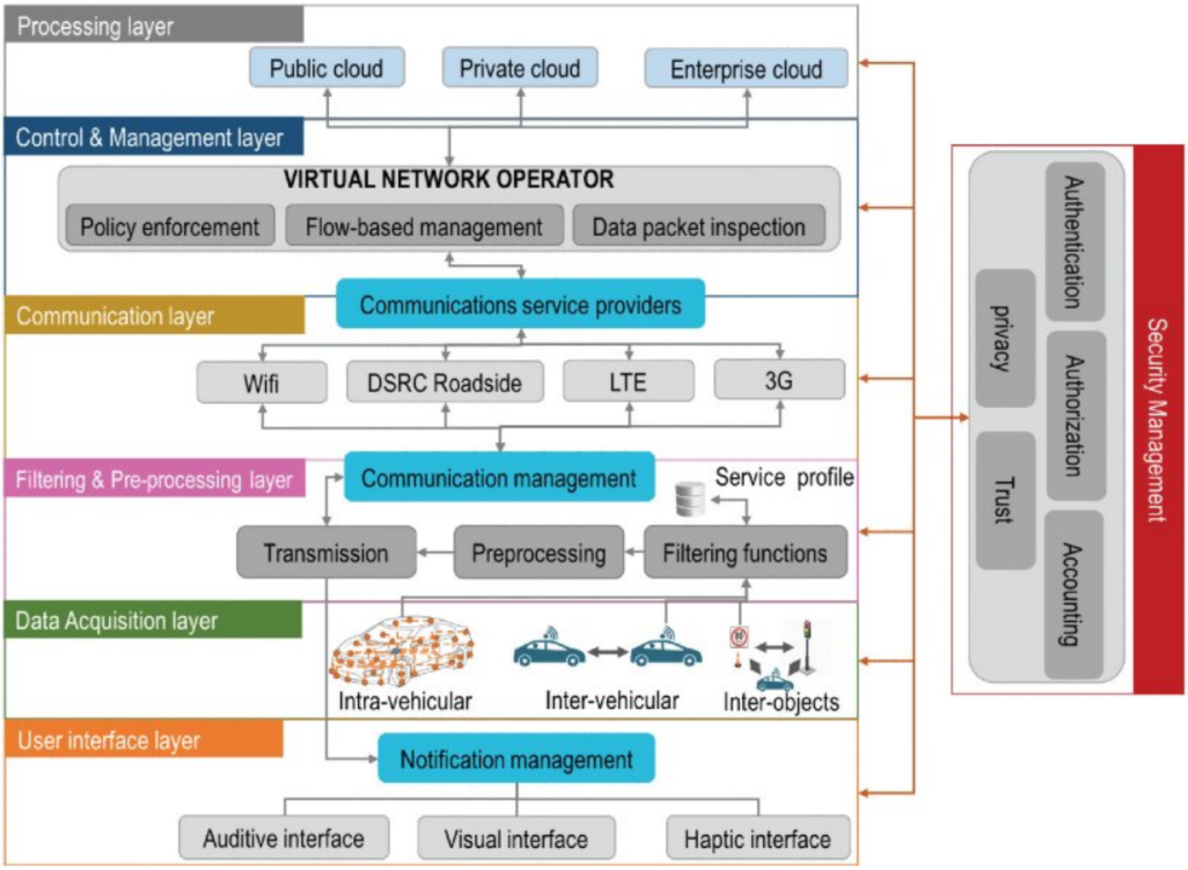


Figure 2.2: IoV architecture [2]

the internet progressed, leading to new Internet requirements. The Information-centric Network (ICN) architecture came to light from these new requirements as an information dissemination alternative to end-to-end communication. ICN is focused on the information exchange between two parties, avoiding continuous connection using a publish-subscribe paradigm [10] [9] [26].

The focus of the ICN architecture is information, and it is through the content name assignment that it is possible to create caching mechanisms that were not previously possible for IP-centric networks. ICN also introduces multicast mechanisms, facilitating the efficient and timely exchange of information between users. However, ICN is much more than the exchange of information between users. New research and developments have emerged regarding information, allowing to address limitations (such as the movement of nodes and the security of information packets) present in the current internet architecture [10] [9] [26].

In this next section, we will focus on the critical concepts of ICN architecture and explain why this new paradigm is so crucial for vehicle networks.

2.2.1 ICN Design

In this section, we introduce the fundamental principles of ICN and a description of the standard design choices for different ICN architectures. This description will set the framework for examining Named Data Networking architecture in detail in the next chapter.

The key functionalities of ICN are:

- Naming scheme:

In ICN, the content-based approach means that the consumer requests content by its name to retrieve from the network, decoupling content from its sources. The way for a user to get the content it desires is by its name, i.e. its identifier, without referring to the node's IP address storing the content. Several decisions need to be made in the design of content namespaces depending on the ICN architecture. Namespaces may or may not reveal structural information. Namespaces can be human-readable and not human-readable. Namespaces can be flat or hierarchical. Namespaces can be self-certifying or not [10] [9] [26].

- Named Data Object:

  The content resources or Named Data Object (NDO) can be chunks of infotainment, services and files. These NDOs keep their identity regardless of how they are copied, stored or transmitted in the network. It is also important to highlight that two NDOs with the same name are identical, i.e., perfect copies. For the NDO to be delivered, the routers use name-based forwarding rules. This approach secures the NDO instead of the transportation channel, which implies that the NDO will be the focus of security measures such as embedded authentication or hash. Since the NDOs alone are location independent, nodes can cache the content, making content available for further network requests [4] [27].

- Publish-subscribe paradigm:

  ICN architectures, such as Content-Centric Network (CCN) [28], Publish-Subscribe Internet Routing Paradigm (PSIRP) [29] and Network of Information (NETInf) [30] are based on the publish-subscribe paradigm [9]. The publish-subscribe paradigm is based on the publisher-subscriber duality. The subscriber uses a name resolution service or a name-based routing service to obtain data published by a publisher. The content sources or publishers make their content available by announcing it to the network. Both operations (publish and get) use the object's name as the main parameter. Consumers, who are interested in a specific type of content, subscribe to the content notification service. This paradigm allows an asynchronous connection between publisher and subscriber, allowing content subscribers to request information whenever they can or need it. The time and space decoupling between publisher and subscriber are fundamental to guarantee timely information delivery to a moving consumer. Furthermore, the publisher does not know how many subscribers are receiving the published content, and neither do subscribers know how many publishers are providing information. In conclusion, the publisher-subscriber paradigm in ICN creates a communication model essential for moving nodes, although different from the traditional connection-based approach [31] [10].

- Caching:

  Caching is a fundamental network mechanism that influences the information dissemination distinction (e.g., content popularity) and availability, scalability, Quality-of-Service (QoS) experienced by consumers, overall network traffic and resource consumption such as overhead and computation costs of nodes. A caching policy model should be considered depending on the network topology and necessities. Caching policies can be categorized by the type, level and nature of the information used to perform a caching decision, for example, node position and content popularity. Caching policies, levels and models differ depending on the ICN architecture. There are two types of caching: on-path caching and off-path caching [26].

  Off-path caching, also referred to as content replication, aims to replicate content within a network regardless of the delivery path(s). When content is requested, the network retrieves it from the source to the consumer, irrespective of the delivery path(s). The nodes can store the replicas along the path depending on certain factors such as node availability, storage availability and content popularity. Furthermore, Replicas are usually advertised in a name resolution service or a named routing service. Depending on the advertisement scheme: The name resolution service handles caches as regular information publishers, or the routing system forwards the requests for information [31].

On-path caching is conducted along the delivery paths, providing the network with the possibility of caching information. On-path caching aims to reduce the traffic and the delay experienced in a network by adding the integration of naming and caching mechanisms into the network layer of ICN networking [31].

- Mobility:

  In nowadays networks, the host-based nature becomes a problem for mobile hosts. ICN does not need an end-to-end connection as it is connectionless and asynchronous, allowing the user to request or send data to a given source without being simultaneously connected to it. The consumer or network will continue to issue requests for NDOs on new access. This request in the Publish-subscribe paradigm works as a request for a new information subscription. In the Publisher case, it becomes more challenging to support mobility since the name resolution, or the network routing tables need to be updated [9].

  The addressing scheme of the internet was designed with fixed hosts in mind, where the exchange of information takes into account each end-to-end location with IP addresses as the key for communication. Time has passed, and mobile users and networks have increased with technological progress. Nowadays, wireless and mobile devices can switch networks by changing their IP address to guarantee communication, but that does not achieve continuous connectivity while on the move. Moreover, communication modes such as Mobile IP bring an inefficient solution for moving nodes [26].

  The Mobile IP protocol, one of the current solutions for moving nodes, imposes ”triangular routing”. A moving user has his requests sent to his home agent (home network) from his current location. The home agent will then forward the request to the desired content source. Afterwards, the source node will send the content through a tunnel to the user. This ”triangular routing” leads to an unnecessary traffic movement since the traffic has a longer path than the optimal, from the source to the destination. This triangular routing can lead to other negative consequences, such as violation of the Border Gateway Protocol (BGP) routing policies for a given traffic destination [26].

  The Publish-subscribe properties of the ICN architectures previously described lead to efficient mobility support and surpass the difficulties originating from the mobile IP protocol. The mobile nodes can reissue subscriptions for information after handoffs, and the network may direct these subscriptions to nearby caches rather than the original publisher [31].

- Security:

  Xylomenos et al.[26] and Ahlgren et al. [9] show that in ICN architectures, security mechanisms come from different key ICN features, such as caching and naming.

  In the context of the publish-subscribe paradigm, content is requested by a user asynchronously, which means that the current internet with end-to-end communication has different security mechanisms. The main focus of security is on the link between nodes, and the security in ICN is focused on packet security. Furthermore, publishers do not need to be aware of the identities of their subscribers since the exchange of packets is provided by the content notification service.

This unawareness is essential since it separates both parties, and information can flow through the network without node authentication.

The naming scheme is a fundamental part of the security mechanisms in ICN. For this reason, ICN architectures implement properties such as self-certifying names. Self-certifying names are responsible for checking the data-name integrity solely based on the data's name. Data-name integrity and, consequently, data integrity are fundamental parts of information security.

Other security mechanisms are provided by caching policies, which are responsible for controlling network congestions and preventing spam attacks such as Denial of Service (DoS).

ICN has different Naming scheme architectures with different naming, mobility, transportation, caching, and security characteristics, and all these architectures aim to distribute content efficiently.

2.2.2 Named Data Networking

Considering the context of vehicular networks in which we have highly dynamic environments with multiple OBUs, RSUs and environment devices exchanging information, ICN and NDN specifically bring improved efficiency and robustness in communication and better scalability concerning information/bandwidth demand.

Amadeo et al. [4] reveals that in a highly dynamic environment such as a vehicular network, the high volume of exchanged information suffers deeply from the poor-quality wireless links and the challenging success in data delivery [27] [32].

Furthermore, the current IP-centric model cannot deal with highly dynamic vehicle movements. Vehicular environments have high mobility nodes, communication ephemerality and intermittency. The rash vehicular environments lead to IP address allocation/maintenance difficulties and stable end-to-end data delivery paths. Changing from an IP-centric network to an Information-centric network (ICN) will allow a paradigm change from end-to-end to content-based communication.

This communication paradigm is important since vehicles are interested in retrieving contents regardless of the identity of the vehicle/roadside node producing it [33]. Furthermore, the network can use the NDN node caching properties to store content and content replicas throughout the network, which is beneficial to the intermittent connectivity on the road [10] [27] [4].

The NDN choice over the other ICN architectures comes from the simple, efficient and essential characteristics of its architecture. NDN focuses on the content name as the critical element in communication. Each content has a unique name that identifies them, i.e., for content copies, every single content has the same name. Furthermore, the content exchange between users no longer focuses on their connection but on the content. Since communication is content-based, the security mechanisms are applied to the node's connection instead of being applied to the content. These features allow each network node to store content in the cache and serve as source nodes for future requests. This architecture thus allows an improved performance in content delivery. Furthermore, this straightforward design of NDN makes him an ideal choice for content-centric networks [10].

The NDN architecture is formally recognized for its hourglass shape, figure 2.3. On the left, we have an hourglass representation of nowadays internet, and on the right, we have the NDN hourglass representation. The hourglass representation originated from the IP global interconnectivity, allowing upper layer and lower layer technologies to transform independently of communications networks. The NDN replaces the thin waist with name objects or content chunks instead of IP addresses or communication endpoints in every node [5].

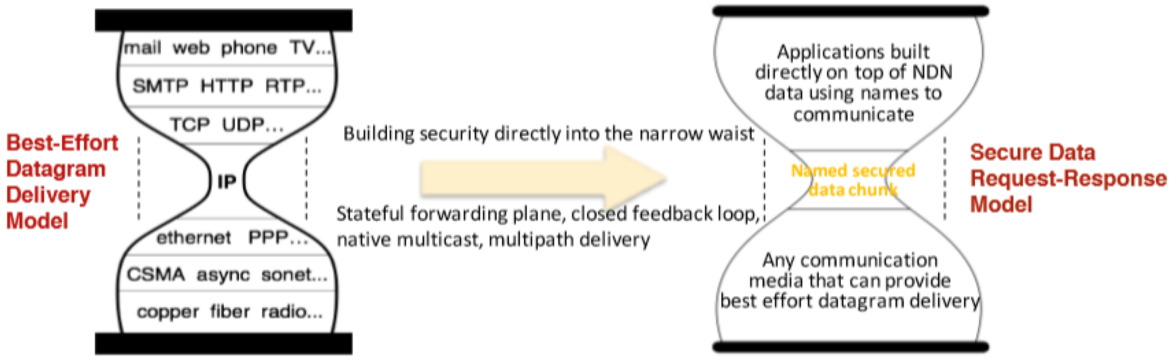


Figure 2.3: Hourglass NDN architecture [3]

NDN System architecture

The characteristics of the NDN architecture guarantee an efficient distribution and dissemination of packets avoiding the addressing difficulties. To avoid addressing, NDN has a name-based packet implementation, which means that the network communication works around the naming scheme of every NDO. There are two types of packets in the NDN architecture, the Interest packet (i.e. the packet used to request a given content) and the Data packet (i.e. the packets with content information). The Data and Interest packets are the terminologies used throughout the work.

To guarantee this efficiency, each node of the network works directly for in-network caching, and its infrastructure is composed of three data structures. The three data structures ensure the functioning of routing and caching mechanisms, which are fundamental for the correct functioning of the NDN architecture. Figure 2.4 shows the flow of the NDN architectures when a consumer receives an Interest and the logic actions and decisions provided by the structures functionality.

The three NDN data structures responsible for the NDN architecture functionality are:

- Content Store (CS):

  CS works as a cache. It is the first data structure searched when a node receives an Interest packet. When a node receives an Interest packet, and the namespace matches the namespace of the Data packet cached, the node transmits the Data packet through the path referenced in the Interest packet back to the consumer node.

- Pending Interest Table (PIT):

  PIT is responsible for storing the incoming Interest packets in a PIT entry until the request for that Interest is fulfilled. The Interest is satisfied by forwarding the content demanded to the consumer node. The PIT entry eliminates the Interest once it is satisfied. Nevertheless, if multiple copies of a pending Interest arrive at a node, aggregation on the same entry name will be made to the PIT.

  PIT is also responsible for storing Interest packet information, on the one hand, to be able to forward the Data packet back to the consumer, and on the other hand, to delete PIT's entry in case its timeout interval expires.

  A node will have different answers when it receives a packet:

  When Data packet arrive, if there is an entry in the PIT, then the Data packet is forwarded to the consumer node, and the PIT is updated.

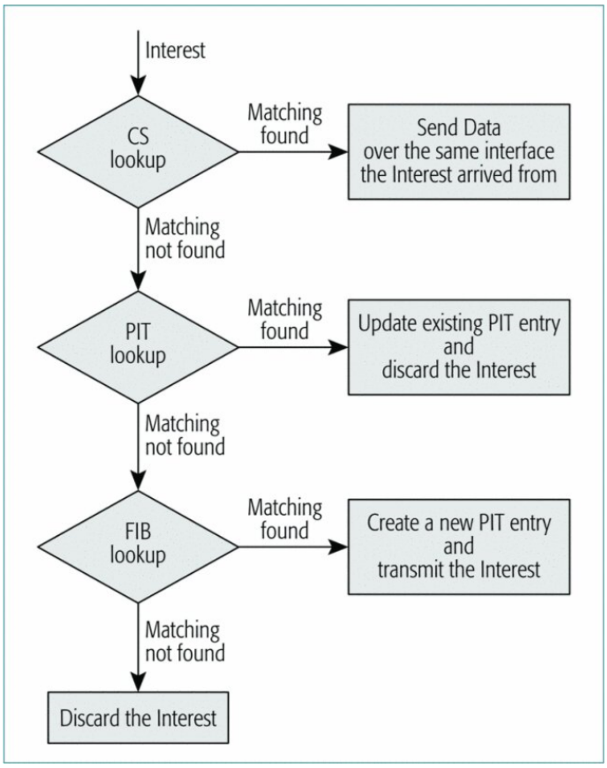


Figure 2.4: NDN architecture flowchart [4]

When an Interest packet arrives with no matching entry in the CS, Forwarding Information Base (FIB) and PIT, then the Interest packet is discarded.

- Forwarding Information Base (FIB):

  FIB stores information regarding next-hop relay nodes to forward Interest packets to a potential source(s) of matching data.

  FIB information is used to forward Interest packets through named-based lookup. In other words, to reach the destination, a next-hop node is selected by matching the longest name prefix with the Interest packet namespace [26].

NDN Packets

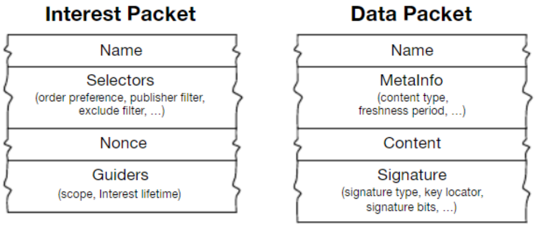


Figure 2.5: NDN message packets format [5]

This section describes the two types of message packets in the NDN architecture.

The namespace of the content requested from the consumer is presented in both Data and Interest Name fields. As shown in figure 2.5, the Interest and Data packet have the Name field in common.

Besides, the Interest packet carries a random nonce generated by the consumer. Name and Nonce are used to identify packets, so the routers which receive the Interest packets can distinguish them from new ones or old ones that looped back (and dropped it) [34]. Signature is as Nonce, a globally unique sequence number responsible for checking duplicate packets. The signature presented in the Signature field binds the producer key with the content. Furthermore, the data producers cryptographically sign every Data packet, ensuring data security, and the publisher's signature (also in the signature field) provides data integrity. The publisher's signature enables the determination of data provenance, allowing a consumer's trust in data to be decoupled from how or where it is obtained [5]. Finally, content encapsulates the content matching the namespace presented in the Name field.

NDN example

Figure 2.6 is an example of the NDN architecture [4]. Consumer C1 needs the information on traffic for highway A3. It issues a broadcast transmission of the Interest packet with the following namespace /traffic/highway/A3/11. After receiving the Interest packet, vehicle R will check in its CS if it is caching the content with that namespace. Because R is caching the content, it will forward the Data packet to the consumer C1. Another example presented in the figure 2.6 is the content request from consumer C3 with the following namespace: /parking/taormina/theater. The C3 broadcasts the Interest packet, which C2 hears. C2 will run the algorithm in the figure 2.4.

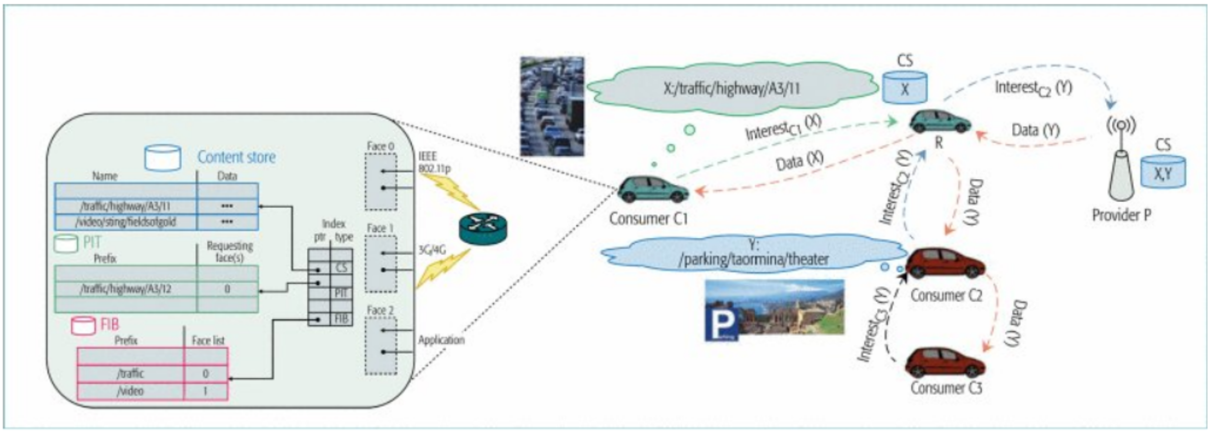


Figure 2.6: NDN architecture example [4]

The algorithm is processed as follows. Vehicle c2 will search its CS whether or not it has the Data packet corresponding to the request. Considering that the CS does not have it stored, C2 will create a field in PIT with details about the received Interest packet. This information allows C2 to forward the Data packet to the consumer. Next, C2 will search its FIB for a reachable destination with a matching name prefix. Being the case (R is a reachable destination with a matching prefix), C2 forwards the Interest packet to R. The vehicle R will run the same algorithm as C2 and thus forward the Interest packet to the data provider P. The data provider will forward the packet to R, checking its PIT. Because the received Data packet satisfies the Interest packet present in the R PIT, the PIT is updated, and the Data packet is forwarded to C2. C2 then returns the Data packet to C3, completing the cycle. Both R and C2 could have stored the Data packet in their respective CS for future requests. If C2 did not have information about a reachable node with a matching prefix in its FIB, it would discard the Interest packet.

Naming

Routers can recognize boundaries between components in names.

Throughout the years, new experimentations and naming designs came to light to facilitate development, and network delivery [5]. However, they attribute no meaning to names. This NDN design decision allows each application to choose the naming scheme that fits its needs, leading to different name design choices.

Current NDN architectures aim for an efficient naming scheme to improve in-network caching, support a scalable network and ensure performance and security in packet dissemination. Nevertheless, creating a naming scheme with such advantages is still a current challenge.

Namespaces divide into the following categories: hierarchical, flat and hybrid namespaces (combination of both).

The hierarchical namespace allows scalable namespaces by adding prefixes that facilitate routing operations. The prefixes can identify applications, timestamps, services, categories and more depending on the namespace design. Moreover, the namespace may have a structure similar to a Uniform Resource Locator (URL).

Flat namespaces apply hash algorithms to the whole or part of the content.

This section analyzes some of the literature regarding different naming schemes. Wang et al. [35] propose a structure of data that is human-readable and follows the pattern: `/traffic/geolocation/timestamp/datatype/nonce`. Analyzing the pattern: the namespace categorizes the NDO under the traffic application. The geolocation identifies the road location, direction and section number. The timestamp represents a period. The data type indicates the data meaning (e.g. vehicle speed). Finally, the Nonce is a random number generated by the publisher to distinguish identical values from different producers. This naming scheme is flexible enough to handle data retrieval at different granularities. Moreover, the data scheme design incorporates data accuracy, duplication detection, temporal and geographical scoping and application variate (i.e. the naming structure should accommodate different data naming structures for individual applications). Nevertheless, Wang et al. [35] named data approach to V2V communications lacks practical proof. Their solution and conclusion, even without results, are worth taking into account for future investigation.

Pesavento et al. [36] propose a naming scheme (/ndn/bit/parking/.../cn) for mobile applications and VANET applications that represents geographic areas and uses an encoding algorithm to leverage the NDN prefix matching mechanism. Unlike Wang, Pesavento et al. [36] evaluated the performance of their proposed naming scheme using ns-3 and ndnSIM simulations. The advantages identified were improved in-network caching, the increased cache hit ratio, and application logic independence from Interest and Data packet forwarding (i.e. no application involved in the forwarding process since the network layer handles NDN names).

Drira et al. [37] propose an NDN Query mechanism (NDN-Q) that collects and aggregates data from a distributed Vehicular network. The NDN-Q extension allows any node to submit a query in the network to collect data. The naming scheme is implemented as a prefix/query and a key/value pair. The key is a node selector (e.g. vehicle, RSU), and the value pair is composed of three fields. The three fields divide into:

- Conditions: Responsible for parameters or attributes of the source node such as position, speed and section;
- Selected key The fundamental values to be included in the data source response, e.g. speed position;
- Time condition Defines the interval and the period between observations;

NDN-Q was implemented based on ndnSIM, and its evaluation concludes that its extension reduces the number of packets collected from data sources, efficiently handles the mobility of vehicles, and delivers query results in a reasonable time.

Quan et al. [38] propose a hybrid naming scheme with three fields: Hierarchical Routable prefix (HRP) provides scalability and aggregation by efficiently routing and forwarding the content. Flat Content Identifier (FCI) that can hash the whole content to use as an identifier of the content chunk and accelerate the content and caching discovery. Primary attribute labels (PALS) store important attribute information for the content, such as priority level, timeliness, caching strategy, etc. The naming scheme benefits each of these fields. It provides optimization strategies such as information regarding the following potential data, which increases streaming services' quality of Experience (QoE).

Caching

The multi-source and in-network caching of NDN help conquer the mobility and intermittent connectivity challenges that are difficult to deal with in traditional IP networks. In-network NDN caching allows intermediate nodes to respond to requests, which relieves the data flow in the network and relieves congestion at the source nodes. These advantages would save not only time in transmitting packets but also bandwidth. However, this is only possible if there is an optimization of caching decision mechanisms and viable data transmission mechanisms.

There are different caching mechanisms, but all aim to improve packet dissemination delivery time, limit and manage network congestion, avoid delays and packet loss and increase packet availability.

Now, We will analyze some of the literature regarding different caching mechanisms and their disadvantages and advantages.

An efficient caching strategy relies on caching valuable content in strategic nodes as it increases content availability and usage in the network. The caching choice considers three main factors: node demand and preference, node importance, and the relative movement of the sender and receiver. Deng et al. [39] propose a design where all nodes within a specific range would receive Data packet, which is different from traditional on-path caching. Deng et al. [39] implemented his strategy design in ndnSIM and compared it with the always caching strategy and the probability of caching of 50% (i.e. P(0.5)). Distributed probability-based caching also reduces the delay and improves the caching utilization by caching necessary data. Nevertheless, when a node prefers popular content over non-popular content, an attacker's spamming of non-popular content will lead to non-popular packet popularisation. This attack has significant security consequences since caching non-popular packets makes caching useless (non-popular packets aren't requested), and the nodes will needlessly waste resources [10].

Quan et al. [38] propose two cooperative caching mechanisms to optimize the Quality of Experience (QoE) of streaming services on a highway VANET scenario. The Partner assisted Cooperation (PAC) revolves around a partner (vehicle in the same lane and at the same speed as the user vehicle) that can help the user vehicle by downloading contents for him and forwarding them later. The PAC provides a high and steady upload bandwidth to its neighbours. The courier-assisted cooperation (CAC) arises when PAC can not provide enough copies of requested contents. Couriers (vehicles in the same lane but with different speeds from the consumer node) have a sporadic connection with the consumer communication area. This caching mechanism can promote content discovery in sparse communication situations due to Couriers' intermittent contact. Although the cooperation is probabilistic, it can alleviate the playback smoothness by predicting desired segments and prefetching them in advance.

Finally, NDN nodes or forwarding decisions require modifications to use cooperative caching schemes, which may violate NDN primitives.

Grewe et al. [40] propose a proactive caching approach for VANETs. The caching scheme starts with an Interest packet sent from the vehicle to a RSU with its position, velocity and Interest frequency, which are used to predict the points in time of the different requests. Besides the information provided by the vehicle, the caching scheme requires additional information such as the data object, the RSU position and the RSU range.

The RSUs position is required for the caching scheme since they are the key to the algorithm and data distribution success. The RSU will transmit the content requested by a vehicle through the network, considering its position and velocity. The algorithm guarantees that the content arrives at an optimal positioned RSU, and the content is cached there. When the vehicle passes that RSU, the content will be transmitted, finishing the cycle. For this reason, the caching strategy optimizes the distribution of data chunks using RSU caching. This scheme does not alter the concepts of NDN architecture. Nevertheless, this scheme neither considers V2V communication nor the variable speed of vehicular nodes.

2.2.3 Routing

This section analyzes some of the literature regarding different routing strategies.

Packet collisions will frequently occur if neighbour nodes rebroadcast Interest packets simultaneously. Blind broadcasting will make Interest packets propagate in all directions and flood the network. If every node forward Interest packets arbitrarily, the network will suffer from unnecessary transmission overhead.

- Geolocation routing:

  Karroubi et al. [41] propose an architecture that aims to reduce packet collision and decrease the error rate to keep the network's Quality of Service (QoS).

  The following scenario demonstrates the architecture functionality:

  A consumer node needs information, so to receive the information, the node broadcasts Interest packets within its communication range. The node(s) that has(ve) the Data packet matching the Interest packet will forward the packet to the consumer within the communication range. Suppose there is no Data packet in the consumer node vicinity. In that case, a relay node (between the middle and the node range extremity) is selected to forward the Interest packet further into the network. Relying on a single node to forward the Interest packet ensures a reduced packet collision, and by avoiding selecting a relay node as the farthest node in the network, this architecture also decreases the error rate.

- Neighbours based forwarding:

  Kuai et al. [42] propose a Density-Aware Delay-Tolerant (DADT) Interest forwarding strategy to retrieve traffic data in vehicular NDN to improve the packet delivery ratio. This strategy relies on the vehicles' temporal and geographical coordinates information to maintain a neighbour list in each network node. The data is crucial to the nodes decide which optimal retransmission and rebroadcast should be made, considering the directional network density.

  The forwarding strategy has two communication phases: the rebroadcast and the retransmission.

  The retransmission: after forwarding the Interest, each node will add the Interest packet to a pending retransmission queue. Before retransmitting, each node will verify if there is a neighbour with higher forwarding capabilities (i.e. the neighbour could transmit to a desirable area). If there is a stronger forwarding node, the node will overhear an Interest packet with the same name and delete the Interest packet.

  The rebroadcast: each forwarding node sets up a timer based on its forwarding priority and adds the Interest packet to a transmission queue. The node with the highest priority will transmit the packet, and the others will cancel the transmission if they overhear a packet with the same name.

Finally, only parts of nodes with higher forwarding priority forward the Interest, mitigating broadcast storms. Furthermore, the retransmission strategy will improve the packet delivery ratio.

- Distributed Forwarding

  Maryam et al. [43] propose a forwarding protocol to overcome the Interest broadcast storm issue and Data packet delivery failure. The proposed mechanism ”IBFS” has the vehicles change the trajectory with their neighbours to resolve these issues. Only one is selected between the consumer node’s neighbours to forward its data. The selection is based on the trajectory (velocity and position). If the course matches the consumer node trajectory, the node is selected as the forwarder. Each node keeps a table with the trajectory of its neighbours. The Interest broadcast storm is eliminated using this method since only one forwarder is selected. The trajectory information allows the Data packet to be sent directly to the consumer on the shortest paths. The proposed scheme performance evaluation proved that IBFS reduces end-to-end delay, the number of Interest packets forwarded and Minimum/Alleviate Interest Satisfaction Delay (ISD).

- Context Aware Forwarding

  Ahmed et al. [44] propose a forwarding scheme ”CODIE” that controls the data flooding/broadcast storm in NDN. CODIE brings attention to the Data packet broadcast issue surrounding consumer Interest broadcast. This issue relates to the nature of the wireless medium and consumer Interest broadcast. When the consumer broadcast Interest packets to the network, multiple neighbour nodes will receive the packets, which will lead to Data packet broadcast back to the consumer, wasting bandwidth and causing congestion and packet loss. CODIE includes a hop count number in each Data packet to eliminate this problem. The hop count number is the number of hops the Interest packet made from the consumer to the provider and is the number of hops allowed to the corresponding Data packet. The hop count number ensures that the Data packet does not go further than the consumer, restricting the number of Data packet in the network.

- Distance aware forwarding

  Wang et al. [45] propose a forwarding scheme that uses distance metrics to avoid the shortcomings of hop-count based metrics and uses an incremental broadcast and adaptive broadcast strategy according to vehicle density to optimize the routing path and reduce packet collision.

2.2.4 Mobility

This section analyses some of the literature regarding different mobility design choices.

Wang et al. [46] proposed a greedy protocol to solve content providers’ mobility difficulties in retrieving data. Moving problems come from moving a node into different domains, which leads to expensive name operations.

The greed algorithm allows the content provider to update the neighbour nodes’ FIB table throughout its movement. Thus, granting the content consumer information about the content provider’s location when it moves. The information updates in the intermediary nodes can improve the communication between the content consumer and the content provider. Nevertheless, the greed algorithm will work side by side with the standard Content-Centric Network protocol and is highly topology dependent.

Yan et al.[47] propose a distributed mobility management scheme based on the content name and vehicle moving information. Yan et al.[47] prove that on-path is helpful for mobility support but can not be implemented alone as it is inefficient in handling fast and continuous request traffic networks. To avoid inefficiency, RSUs cache Data packet in the vehicle’s path by considering the vehicle’s movement and relative position. Nevertheless, RSUs topology, as well as unpredictable vehicle movement, may affect content delivery.

Wang et al. [11] introduce fog computing into information-centric IoV to provide mobility support and avoid continuous FIB updates. The authors focus on solving the consumer and provider mobility problem using fog computing. The fog uses its location-aware ability, storage, and computation support to resolve consumer and provider mobility problems.

In conclusion, in a network of moving nodes, a dynamic and high mobility environment generates issues still challenging to address in the current literature.

2.2.5 Security

In NDN, content security lies in the ”thin waist” of NDN; the name in each NDN packet is bound to packet content with a signature using a key. These keys are used to protect data (or names) via encryption providing data integrity, authenticity and confidentiality. The signature, coupled with data publisher information, enables the determination of data provenance, allowing the consumer’s trust in data to be decoupled from how (and from where) data is obtained.

Furthermore, an application can fetch the appropriate key, identified in the packet’s key locator field, to verify a Data packet’s signature. How to determine the authenticity of a given key for a particular packet in a given application is still a research challenge [5]. The NDN architecture has some benefits over the IP-centric network regarding security since nodes in NDN can not be directly addressable, which means that addressable attacks such as Denial of service or Distributed Denial of service do not work in NDN [10].

Nevertheless, Chowdhury et al. [48] recognize that although there are existing security mechanisms in Data packet, which prevent addressing attacks, there are attacks based on false information dissemination and node tracking that needs to be avoided. Chowdhury et al. [48] imply that authenticating data reduces the chance of accepting false information, but authentication comes with a cost. Authentication relies on the node information to identify it, and that information can not defend the node from tracking. So Chowdhury et al. [48] propose mechanisms that prevent the injection of false information and make it difficult for attackers to track its node.

To authenticate data, a pre-defined trust model verifies the data’s signature. The four-level trust model includes autonomous-vehicle organizations, manufacturers, vehicles and data and relies on ”anchors” (i.e. reputable organizations) to certify keys. These keys can then check credentials and authenticate data. The authors then propose a naming scheme to detect false information. In tracking prevention, they sketched a pseudonym and proxy-based scheme that increases the difficulty of associating different Data packet with the exact producer vehicle.

2.2.6 NDN IoV connection

The NDN architecture has beneficial features in the context of vehicular networks and especially in the IoV paradigm. Characteristics such as name resolution that allows names to be used as means of exchanging content (such as weather conditions, road congestion information, and more) between nodes (eliminating the need for IP-based communication and node identifier) and thus facilitating the exchange of content to vehicles in motion.

On the other hand, in-network caching allows nodes (vehicles and roadside infrastructure) to save content throughout the vehicle network. Thus, allowing easy access to information under the intermittent vehicular connectivity of dynamic moving nodes. In conclusion, the main NDN principles, i.e., name-based communication, in-network caching and architecture, can meet the demand for time and space content retrieval in highly dynamic environments present in the IoV paradigm. Thus, NDN is an obvious solution for efficient information delivery in IoV.

2.3 Producer Mobility

This section presents and discusses the most recent advances in tackling Producer mobility challenges in NDN. First, we define more precisely what are the challenges related to Producer mobility. Then we take a closer look into existing solutions: Mapping Based Mobility Approach (MBMA) [49], Indirection Based Mobility Approach (IBMA) [50], Location-Based Mobility Approach (LBMA) [51] and Control/Data Based Mobility Approach (CDBMA) [52].

2.3.1 NDN Producer Mobility Challenges

NDN architecture supports consumer mobility automatically by using PIT information to send requested Data to mobile consumers. When a consumer relocates to another domain or point of attachment, it resends unfulfilled Interest packets to the content Producer [53].

Nevertheless, Producer mobility remains an open issue in ICN. The Producer, after a handoff, leaves content consumers unaware of its new location. The lack of information on the content consumer side leads to Interest packet loss since the content consumers keep resending Interest packets to the content provider's previous PoA [53]. The lack of content Producer support results in different problems such as Interest packet losses and long handoff delays. In addition, content Producer mobility influences high bandwidth utilization due to frequent routing updates for a large network domain [53] [54].

Serhane et al. [55] introduces Producer mobility problems such as Interest Missing during Handover and Interest Succeed Delivering Before Handover. We describe the Interest missing during handover as the Producer mobility problem we intend to solve.

In NDN, the Interest packet is forwarded hop-by-hop using intermediate nodes' FIB rules to reach the Producer and trace the PIT table path.

However, considering Producer mobility, the Interest packets will follow the FIBs' old path to the Producer without reaching it since the Producer moved, leading to discarded Interests. Therefore, the provider's mobility provokes unnecessary Interest packet losses on the path towards the Producer. Furthermore, the Interest packet loss will continue until the FIBs, which leads them to be dropped, are updated. Thus, until all relevant routers completing the FIB entries are updated, the Producer mobility will result in long handover latency [56] [57].

Lee et al. [50] exposes some consequences on Producer mobility problems. When moving, a Producer requires intermediate routers to update their routing tables to successfully receive future/ongoing content requests. However, updating the routing tables of all content routers requires time and its reconstruction is costly, and inefficient.

These issues have profound implications for the quality of the handover experience and may affect and be unacceptable for time-sensitive applications [57].

- Handover:

  Lee et al. [50] describe the lack of Producer location information after handoff. After the handoff, the intermediate routers need to update their FIBs with the current Producer namespace. When the Interest packets do not reach their destination, the content consumers will unnecessarily send again Interest packets leading to high handoff latency and excessive Interest packet losses [53] [58].

Continuous consumer handover between different access points may affect time-sensitive applications. Consumers may not guarantee a fixed and stable communication session due to the network heterogeneity mobility.

  - Interest Missing during Handover:

    The Interest packet is forwarded hop-by-hop using intermediate nodes' FIB rules to reach the Producer and keeps the path traced in the PIT table. However, the Interest packet will be lost in the network due to Producer mobility phenomena and won't reach the Producer. This issue appears during the handover of the Producer. In this case, we have invalid FIB entries, useless PIT entries, and unsuccessful request/content delivery.

  - Interest Succeed Delivering Before Handover:

    The Interest packet successfully reaches the Producer according to FIB rules before mobility. However, before delivering the content back to the consumer, the Producer moves to another place. In this case, we have valid FIB entries, useless PIT entries, and successful Interest forwarding but content delivery failure [55].

- Prefix aggregation:

  Lee et al. [50] suggest that a plethora of moving Producer nodes leads to routing tables prefixes pollution, which counters the advantages of prefix aggregation. Prefix aggregation is a packet traffic solution to prevent Interest retransmissions that can lead to broadcast storm problems [59] [60].

- Interest aggregation:

  Some features provided by the architecture cannot help mobility support. For instance, Interest aggregation can help reduce the network load by aggregating Interest upstream. Yet, it may lead to loss of content delivery if the Producer moves or the requests arrive in a disjointed manner.

- Cache-Store Mobility Issues:

  Cache-stores are non-control-able entities. Any node can cache the content and move from one network to another. This vehicle behaviour will ignore the benefit of in-network caching. The mobility of cache stores is as essential as the Producers' mobility.

- Name Related Issues:

  The use of location-based names to identify content may not be suitable for mobility. The content name might be updated according to the current path when a node moves from one network to another. This mechanism is not always preferable as the Producer may be active in mobility or can move back to his home network. The use of non-location based names is more recommended.

2.4 Producer based mobility approaches

In this section, we introduce the literature on Producer based mobility approaches. The Producer based mobility approaches are divided in four categories:

Mapping Based Mobility Approach (MBMA) [49] It is a technique that provides mobility support to the Producer using a stable DNS server, rendezvous point or mapping server to track the content identifier and its current provider location, as illustrated in Figure 2.7. The DNS server, rendezvous point or mapping server processes all necessary mappings between the content provider and the content consumer. The Producer sends a namespace update to the DNS server. This forwarding mechanism occurs because the DNS server has permanently stored all the data sources and has its name in all mobile sources. Furthermore, The DNS server name will be the top hierarchical name in all mobile sources. For this reason, all the mobile sources will point and send their interest out to this server. The consumer sends an Interest to the mapping server, which answers with the information in the form of a forwarding namespace hint about the current location of the content provider [6].

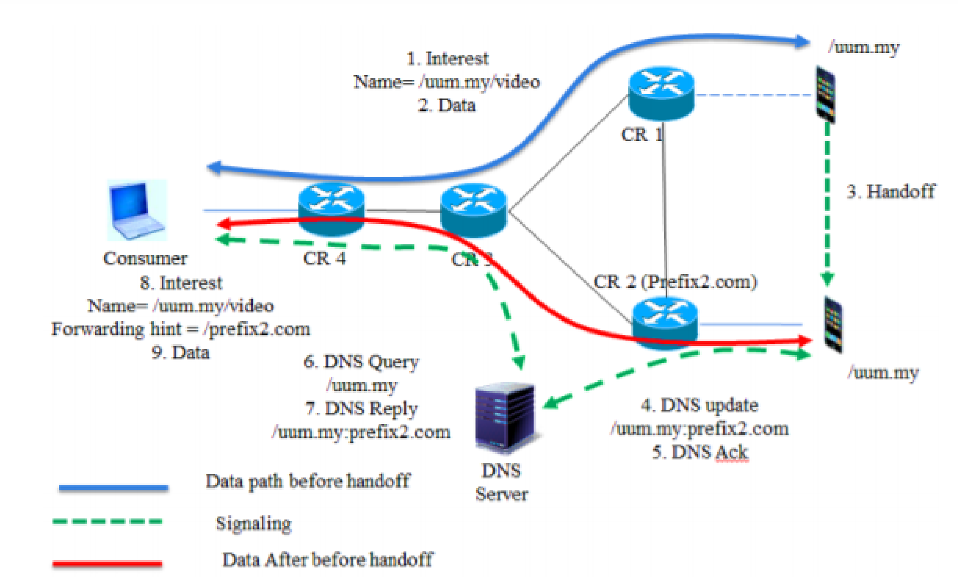


Figure 2.7: Operational Model of MBMA [6]

Gao and Zhang et al. [49] introduce a Scalable Mobility Management (SMM) scheme with three different separation mechanisms to resolve the Producer mobility problem: (i) Content name/location separation, in which case a unique name identifies the content and its location. This separation helps the routing plane during the handover, as the full name of the access router does not need an update. The content name is only used for identifying requested data rather than routing Interest packets. The content source sets the location name of the contents stored locally as the name of the access router. The Producer needs to register the binding between the content name and its location name to the Locator/ID mapping system to support its mobility; (ii) locator/ID separation, in which the SMM scheme uses a Distributed Hash Table (DHT) to create a mapping relationship pair and store it at the mapping servers. The pair comprises a DHT key and a DHT value as the content name hash and the content locator hash, respectively. The objective of the DHT is to support a scalable and robust mapping system and improve the handover process; (iii) management/routing separation, in which case the moving Producer, once it attaches to a new access router, can either send a binding update to the mapping server, updating its new location, or send a binding update to the mapping server and the previous access router, updating both with its new location. This separation allows the Interest packets to be forwarded to the new access router without waiting for a binding update from the mapping system.

Indirection Based Mobility Approach (IBMA) [50] is a technique based upon the Mobile IP mobility solution that uses a home agent to maintain the binding information between the content prefix and the content Producer location, presented in Figure 2.8. The content provider registers its content prefixes with the home router/agent. After the handoff from the current location, the content provider updates the binding information in the home agent, sending a binding update with its new location. The home agent then redirects the Interest packets encapsulating and tunnelling them to the current content provider location. Which conveys that IBMA only uses ICN/NDN between the consumers and the home agent. From the home agent to the Producer, it is used regular IP forwarding [6].

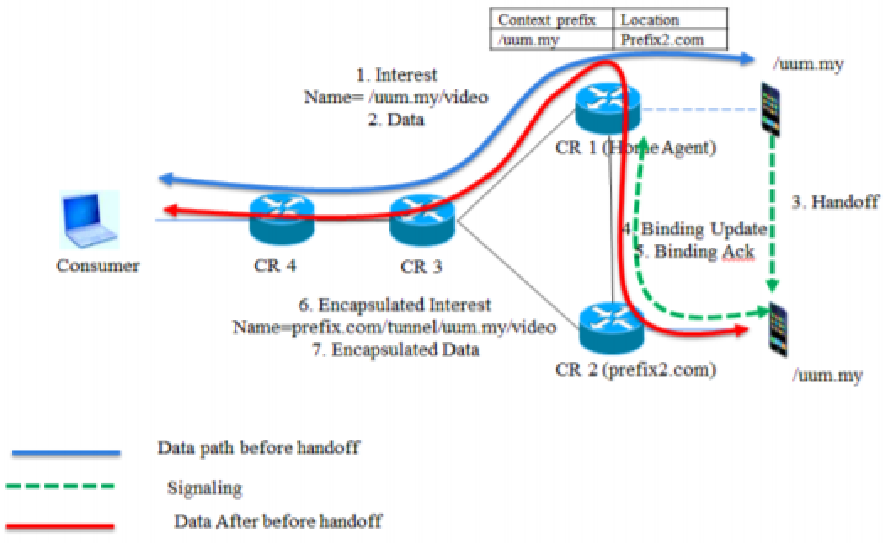


Figure 2.8: Operational Model of IBMA [6]

Lee et al. [50] indirect-based scheme is based on tunnelling Interest packets between a home router (i.e. a home domain content router) and the moving Producer. The Producer, throughout his movement, detects the NDN network attachment point and its name prefix information. The Producer compares the prefix information of the current attachment point with its prefix information. If the prefix information differs, the Producer sends an update to the home router. The home router is responsible for announcing the name prefix of the moving Producer to the other devices in the network. Thus, the network redirects the Interest packets to the home router, which will then encapsulate and send them to the content Producer.

However, Hussain et al. [6], and Lee et al. [50] identify the following problems in the indirect-based approach: (i) Process of encapsulation increases delay; (ii) Addition of extra functionality to the routers and data transmission may result in triangular routing increasing the latency period; (iii) Home router behaves as a single-point-of-failure.

Locator Based Mobility Approach is a technique based on the separation of content identifier and locator, as shown in Figure 2.9. As occurs with the IBMA approach, the LBMA approach relies on the separation between the content identifier and content locator. This separation is also managed by the home router, which is responsible for mapping the content identifier with the content location. The Producer, during the handoff, sends a packet to its home router with an update of its current location. The Interest packets originating from the Consumer are still sent to the home router after the handoff. The home router modifies the incoming Interest packet adding the new Producer location and forwarding the packet to the Producer's current location [6].

Azgin et al. [51] propose a locator-based approach focused on intra-domain and inter-domain anchors. The anchors provide the overlay forwarding functionality as a service for a cluster of mobile flows. The scheme adopts three mechanisms: Anchor assignment, Name resolution and discovery, and Packet forwarding. In the Anchor assignment mechanism, different anchors are assigned to different namespaces using hash-based namespace groupings. Each ICN router within each domain is assigned unique hash-output identifiers. This assignment allows each ICN router to determine the name anchor responsible for a given namespace within its domain. With the name resolution and discovery, new Producers register into their local name anchor or anchors through a local name resolution (mapping the namespace hash to determine the local name anchor). The Producer is finally registered when the

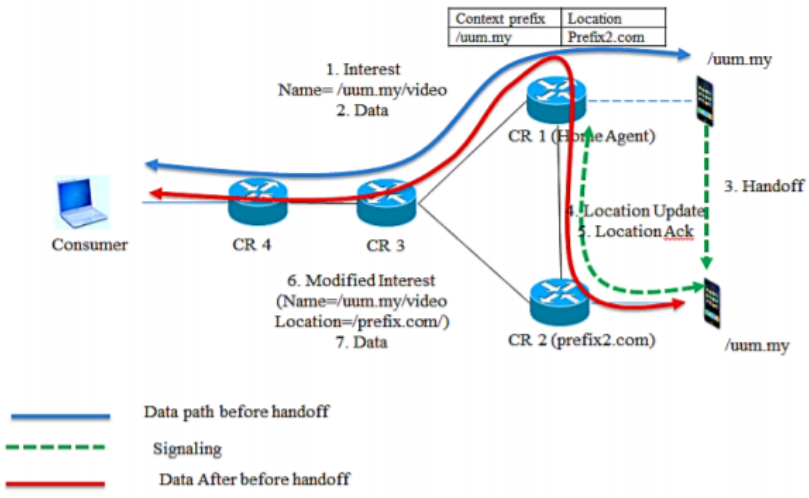


Figure 2.9: Operational Model of LBMA [6]

local name anchor registers the Producer's namespace at the corresponding global name anchor. The packet forwarding mechanism is responsible for label forwarding from both the Consumer and Producer side. The proposed architecture uses label-based forwarding to route packets to their corresponding namespaces anchors.

Together these mechanisms are responsible for handling the Producer mobility with reduced latency in both inter and intra-domains. However, the proposed mechanism integrates three new structures into the existing NDN architecture to manage name resolution, discovery and packet forwarding.

Control/Data-planed Based Mobility Approach is a technique based on the control plane and data plane separation to support mobile Producer, as illustrated in Figure 2.10. The control plane uses a server that controls the mobility signalling process during handoff. After the handoff, the data plane is responsible for forwarding packets between Consumer and the Producer. During Producer handoff, the Producer sends a location update to the controller [6].

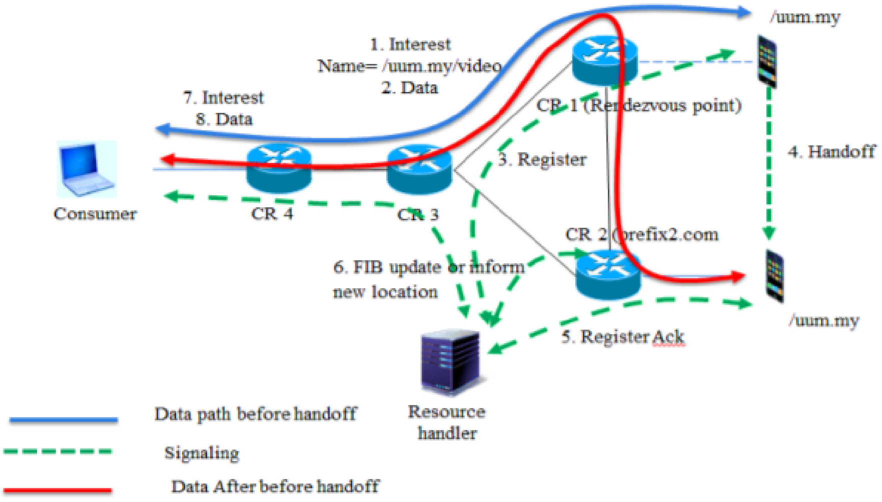


Figure 2.10: Operational Model of CDBMA [6]

Torres et al. [52] propose a Controller-based Routing Scheme ”CRoS” that follows the Software Defined Networks (SDN) technology. The CRoS is divided into two phases: A Bootstrap phase and a Name-Data routing phase. The former is responsible for the construction of the global view of the network. The controller receives information from all routers to create a global network view: i) registration information (routers send information about their presence and neighbours to the controller; ii) control discovery information (routers broadcast Interest packets to find the controller). In the Name-Data routing phase, a new Producer starts by registering itself by sending a specific Interest packet. The Producer sends the Interest packet directly to the controller, which registers the new Producer (stores the Producer content prefix in the Named-Data location table). This phase creates end-to-end routes between Consumers and Producers with a single route request. The two phases allow the controller to compute the best routes for content delivery.

The CRoS approach aims to reduce the routers-controller communication overhead by: (i) coding routing information on content names; (ii) reactively updating the controller upon router local information change; (iii) avoiding the replications of routing information from the controller to routers; (iv) restricting the Interest flooding. However, the CRoS protocol has decreased efficiency when the Producer mobility rate increases due to additional Interest packets required for the features mentioned above.

The analysis of the approaches described in this section should address the challenges posed by the presence of a mobility anchor, such as a DNS server (MBMA), the home router (IBMA and LBMA) or a resource handler (CDBMA). Moreover, while anchor-based approaches seem suitable for fixed networks, their management may face additional challenges in IoV networks characterized by their ad-hoc networking nature. Therefore, the best option is to use a proposal that does not rely on an anchor to manage Producer mobility. Such anchor-less-based mobility management proposal should rely only on the usage of a signalling protocol based on Interest packets to directly update the FIB in any needed node in the data plane, supporting Producers' mobility even in the presence of latency-sensitive applications.

2.5 Anchor-less Mobility Management

Anchor-less-based mobility-based management allows mobile nodes to advertise their mobility to the network without requiring any specific node to act as a Rendezvous point. In this section, we take a closer look into Map-me [7], an anchor-less approach. Map-me [7] is based on the following characteristics: (i) it addresses Producer mobility while focusing on providing fast handover and preserving progressing flows performance; (ii) it relies upon propagating forwarding updates of the Producer position to guarantee Producer mobility reliability.

The main contribution of Map-me as an anchor-less mobility-management mechanism is to leverage stateful forwarding and dynamic and distributed Interest load balancing to update the forwarding state at routers. Compared to the described anchor-based solutions, Map-me provides: (i) a simple data plane approach avoiding additional name resolution and discovery complexity; (ii) does not need a third party (a controller or rendezvous point) to process mobility handoff and location control; (iii) is lightweight in terms of required signalling messages; (iv) can better distribute traffic over the network and thus better cope with a more significant number of users. The Map-me solution is, for this reason, a potential anchor-less mechanism to solve the Producer mobility problem in IoV.

The Map-me solution comprises two components: an update protocol 2.11 and a notification/discovery protocol that complement each other.

The update protocol uses an Interest packet called Interest Update (IU), which the Producer forwards every T seconds to its previous FIB location. A special flag in its header identifies the IU. All routers that relay the IU packet from the current Producer location into the Producer's previous location update their FIBs, including the previous hop that forwarded the IU packet as the next hop towards the name prefixes

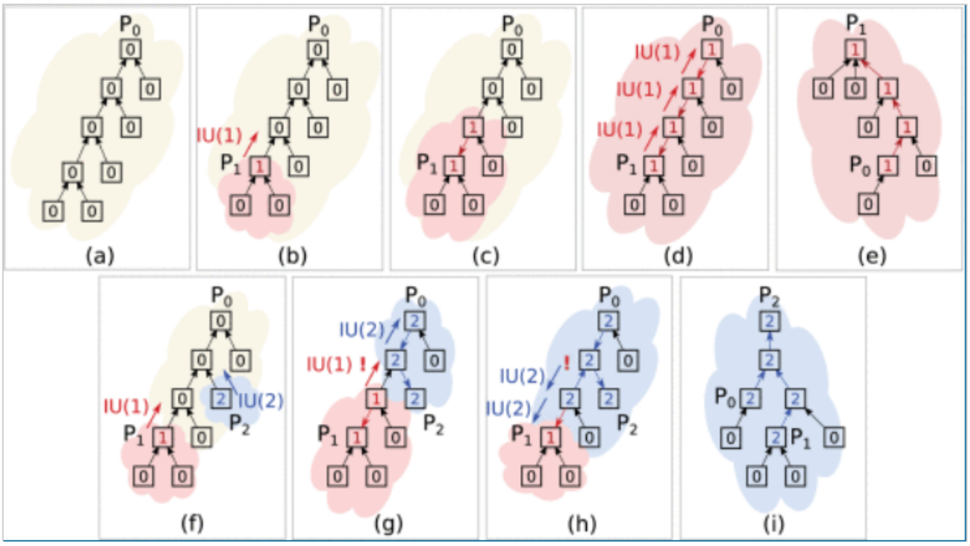


Figure 2.11: Map me IU mechanism [7]

offered by the Producer.

The discovery protocol is divided into the Interest Notification (IN) mechanism and the Producer Discovery (PD) mechanism.

The Producer uses the IN mechanism to ensure frequent yet lightweight Interest Notification packet broadcast to the network. The IN differs from the regular Interest packets by adding an identification flag and a sequence number to its header. The sequence number is used to modify FIB entries. The sequence number triggers FIB modification when it receives an IU that carries a higher sequence number than the one locally stored. The Producer uses the IU to follow FIB indication and update FIBs Tables through the network. However, contrary to the IU, the IN is broadcasted with one hop limit to all Producer neighbours. The Producer sends IN packet every time it attaches to the access point. The IN routes the Consumer requests to the Producer before the IU protocol finishes since the IN has only one hop distance.

The PD protocol is used when a Consumer Interest packet reaches a point of access with an invalid output face in the corresponding entry. After reaching this point, the Consumer Interest packet is broadcasted to a one-hop distance to all neighbours, aiming to find the breadcrumbs (IN) left by the Producer mobility. Therefore, the Consumer Interest packets can be forwarded directly to the Producer or be broadcasted if the point of access has no valid output face.

To implement such protocols, Map-me modified its FIB to incorporate additional information: sequence number and an associative array. The sequence number is used to prevent forwarding loops, control the propagation of Interest packets by the PD mechanism, and handle concurrent updates. The associative array maps a face on which IU has been sent with the associated retransmission timer.

Although Map-me seems a suitable solution to implement an anchor-less mobility management mechanism in a fixed network with the help of an existing routing protocol, it would face several challenges in a dynamic network scenario such as the one posed by IoV. These challenges include frequent connectivity disruptions, durability, and sudden network changes caused by node mobility and urban obstacles [61].

The MAP-me base operation relies on the Interest Update packets, which follow the FIB populated by the routing protocol. In IoV, the IU induces significant delays or fails due to constant network change.

The Map-me proposal was not created for a highly dynamic network without a routing mechanism. Although there are name-based routing protocols for fixed networks (e.g. Open Shortest Path First

(OSPFN) and Named-Data Link State Routing (NLSR)), there are no name-based routing for dynamic networks [26]. The MAP-me base operation relies on the Interest Update packets, which, besides following the FIB populated by the routing protocol, induces significant delays. PDCN (Producer Data Controlled Notification) is a named-based routing protocol created to solve the challenges presented by IoV. PDCN uses its neighbour degree and Interest Notification method as routing and forwarding mechanisms to ensure data delivery while avoiding Interest broadcast problems without a routing protocol.

2.6 Simulator

This section presents different simulators needed to create, simulate and analyse the producer mobility solution in an IoV scenario. First, we explain that SUMO is a helpful traffic simulator which uses traffic management solutions and accurate knowledge of the traffic conditions and vehicle dynamics to expose our producer mobility solution [62].

Next, we present an overview of the ndnSIM (NDN simulator based on the NS-3 [63] simulation framework) design. The ndnSIM has a variety of NDN forwarding and in-network caching functionalities, which facilitate the development of different NDN applications. The ndnSIM also has different network environments to choose from, different routing protocols and different designs of congestion control [3].

ndnSIM is implemented as a new network-layer protocol model and can run on top of any available link-layer protocol model (e.g., point-to-point, Carrier Sense Multiple Access (CSMA), wireless). In addition, the simulator provides an extensive collection of interfaces and helpers to perform detailed tracing behaviour of every component and NDN traffic flow.

2.6.1 SUMO

SUMO is a traffic simulator with the necessary simulation tools from real-world detectors to create advanced and precise traffic models. Sumo uses real-world elements like network data (e.g. roads), traffic infrastructures and demand, and macroscopic and microscopic models to create complex vehicle models and real-world traffic dynamics [62]. Sumo also provides a visual representation of the traffic and vehicle dynamics simulations, which is very useful for qualitative validation. The simulation extracts many output files like vehicle trajectory position and speed, traffic data aggregated over network elements (lanes or edges) and traffic data aggregated for the whole simulation. These files are all necessary to create real-world vehicle scenarios to best demonstrate the reliability of our producer mobility solution in VNDN [62].

We generate multiple scenarios from different selected areas from the world map provided by Sumo. We set some parametrized traffic modes to simulate daily traffic demand to the selected map network. Since traffic simulation models are typically used for stochastic behavior, we use SUMO to simulate different scenarios a number of times and draw statistical conclusions.

Sumo has the tools to recreate IoV scenarios and apply the NDN and PDCN architecture created in ndnSIM over the developed environment.

2.6.2 NDN Forwarder

The ndnSIM includes the NDN forwarder, which consists of the NDN architecture with its three data structures (FIB, PIT, CS), and is responsible for core NDN operations. NFD enable easy experiments with new protocol features, algorithms, and applications for NDN.

2.6.3 NS-3 Framework

The NS-3 framework is a network simulation platform that can create simulation topologies with custom node and link parameters. The NS-3 framework can also trace and collect simulation data and visualize the execution. In NS-3, the network simulations are based on the following key abstractions:

- Node, the basic computing unit, which can be programmed;
- Application, where different functionalities can be tested;
- Net device, representation of both the simulated hardware and software drivers that enable a Node to communicate through Channels with other Nodes;
- Channel, the communication channel between nodes;
- Topology helper, a set of software components that configure the previous abstractions. The software is organized with several modules (e.g., Network, LTE, Antenna, and Wi-fi), models (e.g. Internet, mobility, propagation, Ad Hoc On-Demand Distance Vector) and several helpers classes.

2.6.4 ndnSIM structure

The ndnSIM structure includes the implementation of the NDN forwarder (i.e., Named Data Networking Forwarder (NFD)), the NS-3 existing modules, models and integrated components, and the ndn-cxx, Figure 2.12. The ndn-cxx library allows ndnSIM to simulate real-world NDN applications. Ndn-cxx implements the major NDN primitives used to implement various applications. The ndnSIM comprises the following components: Core (Integration and Models), Utilities and Helpers, each with different features.

- Core: the NDN protocol stack, the realization of NFD's Transport to provide communication on top of NS-3 NetDevice and Channel abstractions, the realization of NFD's LinkService to facilitate direct communication between ndnSIM specific applications and local forwarder instances, and the global routing controller to facilitate static configuration of FIB (based on the Dijkstra's shortest path algorithm).
- Utilities: Several packet tracers to obtain simulation results (link-level, network-level, and application-level tracing) and topology readers simplify the definition of simulation topologies.
- Helpers: Components used to install and configure the ndnSIM network stack, manage FIB, forwarding strategies, and cache replacement policies, and simplify modifying states (up/down) of the links in the simulated topologies (used in an example in the section below).

2.6.5 ndnSIM Applications

ndnSIM has two types of applications:

- Specific applications: Responsible for generating Interest/Data packet flow for different situations based on NS-3 applications. Specific applications such as the ConsumerCbr (Generates Interest traffic at a constant frequency, i.e., acts as a consumer node requesting Data packets) and the producer (responds to the received Interest packets with the Data packets carrying the same name as the Interest packet and with a specified size) are used in a simple example described in the section below;

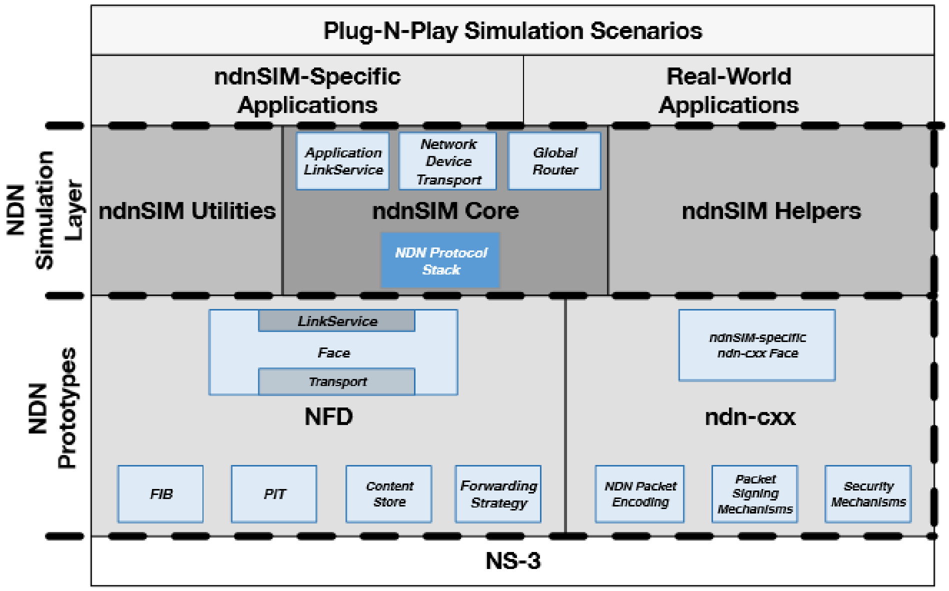


Figure 2.12: Structure of the ndnSIM simulation package [3]

- The real-world applications: ”generic applications and libraries that fully leverage the high-level NDN and asynchronous input/output APIs provided by the ndn-cxx library”. An example of a ndnSIM real-world application is NDN Link State Routing Protocol, a routing protocol for NDN that propagates reachability to name prefixes instead of IP prefixes.

Finally, ndnSIM is an open-source simulator which allows public users to download and modify the codebase. The liberty to change the codebase will enable us to use the ndnSIM in different applications and provide models, functionalities, and architectural characteristics to apply NDN in a real-world application such as IoV.

For example, Deng et al. [39] use ndnSIM to implement a changed NDN architecture and add novel functions and compare their new NDN design with the traditional NDN.

2.6.6 NDN simulation example

This section introduces a simple example of three nodes connected with point-to-point links, one NDN consumer and one NDN producer, using the NDN simulator, as shown in figure 2.13.

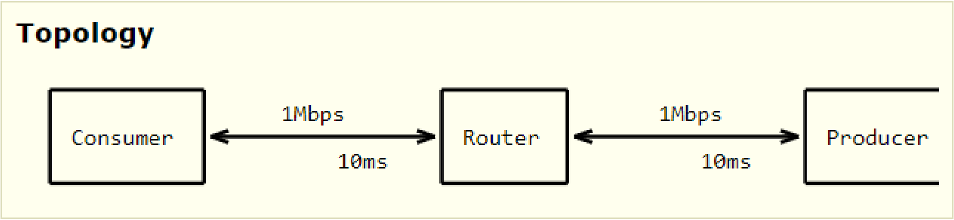
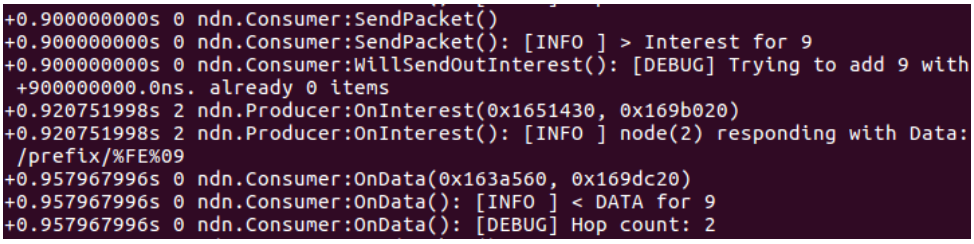


Figure 2.13: NDN simple example topology

The ConsumerCbr simulates the consumer application that generates Interest packets towards the producer with a frequency of 10 Interest packets per second with the following namespace ”/prefix/x”, being x the number of the Interest packet sent (e.g. ”/prefix/0” for the first Interest packet).

The Producer class simulates the producer application and satisfies all incoming Interests with virtual payload data (1024 bytes). The producer will reply to all requests starting with ”/prefix”.



```
+0.900000000s 0 ndn.Consumer:SendPacket()
+0.900000000s 0 ndn.Consumer:SendPacket(): [INFO ] > Interest for 9
+0.900000000s 0 ndn.Consumer:WillSendOutInterest(): [DEBUG] Trying to add 9 with
 +900000000.0ns. already 0 items
+0.920751998s 2 ndn.Producer:OnInterest(0x1651430, 0x169b020)
+0.920751998s 2 ndn.Producer:OnInterest(): [INFO ] node(2) responding with Data:
 /prefix/%FE%09
+0.957967996s 0 ndn.Consumer:OnData(0x163a560, 0x169dc20)
+0.957967996s 0 ndn.Consumer:OnData(): [INFO ] < DATA for 9
+0.957967996s 0 ndn.Consumer:OnData(): [DEBUG] Hop count: 2
```

Figure 2.14: Simple example simulation output (ndnSIM)

Figure 2.14, represents a fraction of the three nodes topology simulation. The consumer sends the 10th Interest packet as the simulation will reach 1 second. After receiving the Interest packet with the name: ”/prefix/9”, the producer responds with the respective Data packet. The Data packet then travels throughout the network (in this case, there is only one node between the producer and the consumer) and reaches the consumer following the routes established in the FIB entries of each node (managed by the FIB helper).

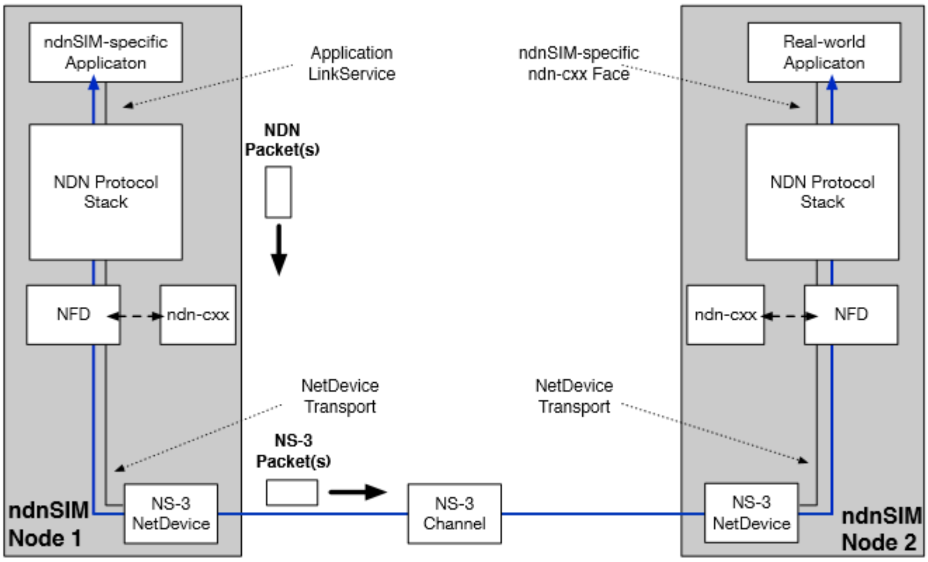


Figure 2.15: Packet flow between two nodes ndnSIM architecture [3]

Figure 2.15 represents the packet flow between two simulated nodes. This packet flow involves NS-3’s packet, device, and channel abstraction, ndnSIM core, and processing by the integrated NFD with the help of the ndn-cxx library.

We will use this simulator to implement and design our NDN architecture and put it to the test in real-world applications presented by the IoV paradigm.

Chapter 3

Work proposal

The main goal of the following work is to develop an efficient information-centric mechanism integrated with a anchor-less approach to ensure the content producer mobility support in the context of the next generation vehicular network.

We propose PDCN (Producer Data Controlled Notification), which intends to solve the producer mobility problem through the data plane. The PDCN is a forwarding and routing mechanism to solve the producer mobility problem while assuring high packet delivery success, Interest broadcast minimization and low packet delivery delay in a highly dynamic network.

PDCN uses its neighbour degree and Interest Notification method called Notification Controlled Broadcast (NCB) as routing and forwarding mechanisms to ensure Data delivery while avoiding Interest broadcast problems.

The degree method is each node's decision to broadcast the Interest Notification considering the number of existing neighbours in its vicinity. Each node has a neighbour list, a list with the nodes that, during a given period, the node interacted with. The neighbour's list length is the degree value. This value constitutes part of the IN packet.

During communication, each node updates the neighbour list with the nodes it enters into contact with. During the Interest Notification broadcast, every node receiving the IN will analyse its neighbour list and compare its length with the Interest degree number. Each IN has a degree number. After analysing the values, the node updates the degree value and broadcasts the Interest it received if it has more neighbour nodes than the node from which it received an Interest. Otherwise, the node does not broadcast the Interest.

The NDN architecture has a pre-routing mechanism that populates every Consumer node in the network with FIB-updated information by default, which is the necessary information to request content. Without pre-routing information, Consumer nodes discard the Interest packets. For this reason, NDN does not broadcast Interests without pre-routing information because it has no FIB next-hop information to request content.

However, PDCN is a Producer and a routing mobility solution. In scenarios where pre-routing information is not predefined, like in IoV environments, PDCN solves the routing problem by Updating FIBs with fresh next-hop relay nodes with a Data source and providing this information in a controlled manner through the network. The Producers use IN to inform the network of its movements by updating the FIBs. The IN updates the FIBs allowing the nodes receiving the IN to forward the Interests to the Producers using a specific channel, avoiding broadcast.

Our proposal uses a new version of the Interest notification mechanism of Map-me with a modified set of features. Furthermore, our mechanism does not use the Interest update mechanism nor the discovery mechanism applied by Map-me. While Map-me depends on the existence of a routing protocol that populates the FIB, our proposal only uses our new Interest notification mechanism to populate nodes' FIBs. The degree method provides a controlled broadcast of the Interest Notification packet, not one hop as in Map-me Interest Notification broadcast but multi-hop. Map-me implements its mechanism in a fixed network, while PDCN does not. The difference between fixed and mobile networks impacts how the packets, IN, are used. Map-me uses IN as a breadcrumb left by Producers at every encountered PoA. In PDCN, the IN is broadcasted periodically by Producers.

3.1 ndnSIM and PDCN

In this section, we look at the implementation of PDCN in the ndnSIM. First, we delve into the specifications of the simulator's data plane process since PDCN solves the mobility problem through the data plane. Then we showcase the differences made to the data plane process to implement PDCN.

ndnSIM data plane involves the NDN Forwarding Daemon (NFD) packet processing part, called forwarder. NFD is a network forwarder that implements the NDN protocol. The packet processing in NFD is composed of forwarding pipelines and forwarding strategies.

We focus our approach on the forwarding pipelines. The forwarding pipeline is a series of packet or PIT entry processing steps. The processing steps are triggered by a specific event (e.g. incoming Interest or outgoing Interest).

The forwarding pipeline operates on the network layer packets (Interest and Data), and each packet is passed from one step to another until all processing is finished.

PIT, CS, FIB and StrategyChoice tables information are used to process the packets. FIB and StrategyChoice tables are managed by their corresponding managers and are not directly affected by the data plane traffic.

Furthermore, the pipeline has direct access to the face table. The Face table has all active faces in the forwarder. A Face is an abstraction that implements communication primitives to send and receive Interest and Data packets. In other words, a Face is responsible for sending and receiving Interest and Data packets through a protocol-specific tunnel.

NFD separates the Interest processing in different pipelines. We first take a closer look into the incoming Interest pipeline operated by NFD. Secondly, we explain the modifications and main differences made to the NFD incoming Interest pipeline to create our approach.

NFD incoming Interest pipeline is implemented by the OnIncomingInterest method, which is triggered by a Face::onReceiveInterest signal, i.e., the Interest packet received by the Face triggers a signal to process the newly received packet.

After checking if malicious users send the packet, the next step is looking up existing or creating a new PIT entry, using the name and selectors specified in the Interest packet. The PIT creation before CS lookup is to reduce lookup overhead. CS is most likely significantly larger than PIT and can incur significant overhead.

The next step is checking Interest Nonce against Nonces in the PIT entry and Dead Nonce List. The Nonce is a randomly generated number created, which is used to detect looping Interests. If a match is found, the incoming Interest is considered a duplicate due to either loop or multi-path arrival. Thus the packet is dropped. If a match is not found, processing continues. The PIT entry timers are cancelled because a new valid Interest arrived for the PIT entry, implying the PIT entry lifetime extension. After, the pipeline tests whether the Interest is pending, i.e., the PIT entry has already another in-record from the same or other incoming Face.

If the Interest is not pending, the Interest is matched against the CS. Otherwise, CS lookup is unnecessary because a pending Interest implies that a previous CS returns no match. The next steps applied in the OnIncomingInterest depend on the Interest CS miss or CS hit.

The PDCN forwarding pipeline operates on the network layer packets, and each packet is processed using PIT, CS, FIB and StrategyChoice tables as forwarding NDN pipeline. The difference between our approach pipeline resides in the FIB being affected by the data plane traffic, while the NDN forwarding pipeline does not.

After the first step, i.e., checking if malicious users send the packet, the next step is analysing the Interest packet header. If the Interest is not an Interest notification packet, the OnIncomingInterest method described before is applied. If the Interest received is an IN, further steps are applied if it arrives at a Consumer or a Producer. The IN is checked against the Nonces in the Dead Nonce list to avoid IN loops or multi-path arrival in both cases. Then the IN name and Nonce are added to the Dead Nonce List.

A new FIB entry is created if the IN arrives at the consumer. The Interest packet name, outgoing Face and routing cost is stored in the FIB entry. The FIB entry associates the name prefix to a Face. The Face can reach a potential source of matching Data with the same prefix. A timer is set to establish a FIB entry lifetime. After the timer timeout, the FIB entry is removed from the FIB. The next step for the consumer is to create a new IN packet with its number of neighbours element modified with the respective Consumer neighbour node (NN). Next, the pipeline decides to broadcast the new IN to all available faces in the face table if the NN on the Interest received is less than the Consumer NN. If not, the pipeline does not broadcast the new IN.

This increased support in mobility is one of the main challenges of producer mobility in NDNs. The increase in mobility efficiency, reliability and possible scalability will positively impact the performance of the vehicular applications, which is one of the end goals of this work proposal.

We will implement this approach in the NDN simulator (ndnSIM), evaluate different test cases containing different scenarios, and compare the results with the NDNs mechanism.

3.2 Simulation scenarios and parameters

PDCN implements its scenario in the ndnSIM environment. Our base scenario runs for 200 simulated seconds with 40 nodes (i.e. vehicles) moving according to a realistic vehicular mobility pattern using SUMO.

Sumo has the tools to recreate IoV scenarios and apply the NDN and PDCN architecture created in ndnSIM over the developed environment.

The simulated vehicles ”run” within the road grid section of Manhattan, New York, and travel based on models derived from actual traffic data. PDCN uses IEEE 802.11p WiFi on 5.9GHz frequencies, ITU-R 1411 LOS propagation model, and a 10 MHz Control Channel (CH) for all traffic. All Consumer and Producers nodes transmit Interest and Interest notification packets with a uniform distribution frequency.

The application considered was a streaming audio/video application characterized by a CBR rate of 1Mbps with retransmission in the baseline scenario. We take two approaches to consumer request patterns. In the first approach, consumers request packets according to a uniform distribution. The second consumer request approach, according to content popularity, each consumer requests packets with realistic request patterns modelled using Zipf-Mandelbrot distribution. We use the Least Recently Used (LRU) cache policy for object caching, and multicast is used as the forwarding strategy when FIB has updated information [64].

We study and analyse node density, frequency and Consumer and Producer variation and measure their impact on PDCN and NDN. The results provide essential information regarding the content replication and delivery time in a moving vehicular network and the PDCN performance against NDN.

Finally, we test PDCN with different consumers and producers to measure the impact of Interest and IN replication in the network and its consequences.

3.3 Performance Metrics

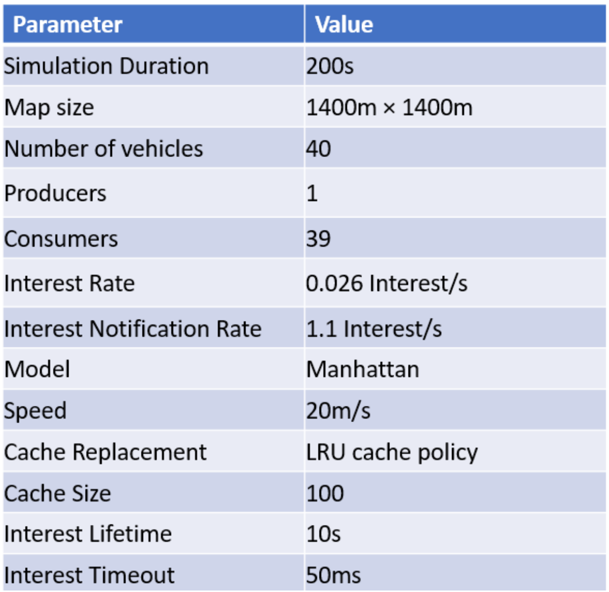
The following Performance Metrics were selected to provide a better understanding of the scheme performance.

- Success Rate: Proportion of successful Interest delivery, i.e. the proportion of successful delivered Interest by total Interest generated;
- Last Consumer Delay: Represents the delay between last Interest sent and Data packet received;
- Full Consumer Delay: Represents the delay between first Interest sent and Data packet received (i.e., includes time of Interest re-transmissions);
- Hop Count: Hop count is the number of network hops that the retrieved Data packet traveled on the way back from producer application or cache;
- Retx Interests: Number of Interest retransmissions;
- Cache Hits: The Number of times an Interest is Satisfied by a Data cached;
- Cache Misses: The Number of times an Interest is Unsatisfied by a node;
- Overhead: Calculated as the percentage of total number of control packets generated by the scheme to the total number of Interests sent;

Chapter 4

Results

This section presents the aforementioned mobility management scheme implemented and results. The details of the experiments and select results are shown and discussed.



Parameter	Value
Simulation Duration	200s
Map size	1400m × 1400m
Number of vehicles	40
Producers	1
Consumers	39
Interest Rate	0.026 Interest/s
Interest Notification Rate	1.1 Interest/s
Model	Manhattan
Speed	20m/s
Cache Replacement	LRU cache policy
Cache Size	100
Interest Lifetime	10s
Interest Timeout	50ms

Table 4.1: Simulation values

Table 4.1 summarizes the simulation parameters used for the first simulations in order to test the PDCN implementation performance against NDN.

The simulation parameter values create a more straightforward, accessible and fast simulation result to develop and analyse. The initial results created a framework case to compare the new implementations easily.

The Consumer Interest frequency was selected to avoid an Interest broadcast storm since the IN (Interest notification) frequency, although 42 times higher than the Interests frequency, is only sent by one Producer and does not have a re-transmission mechanism as opposed to the Consumers Interests. In a simulation with only one Producer, it takes time to mobilize its content through the network. All vehicles have different movement patterns, and it will be challenging to guarantee interactions with all vehicles, i.e. more Interest packets will be lost at the beginning of the simulation.

In NDN, the Producer will disseminate its content with every node he enters in contact with. Each Consumer interacting with the Producer will cache its content. The Producer and Consumer for the base scenario will only transmit/request a payload with 1024 bytes.

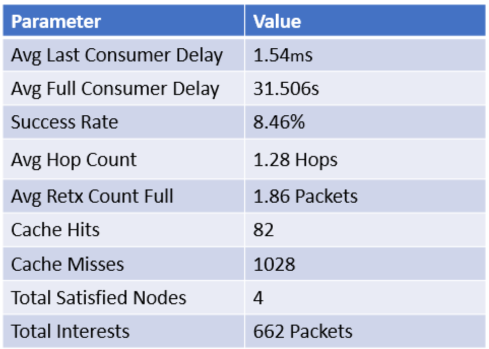
For this reason, it is expect that NDN implementation will lead to high-Interest loss and unsuccessful Data broadcast through the network, i.e. fewer Consumer vehicles will receive and cache Data packets.

However, PDCN Consumers interacting with the Producer will cache its content and disseminate the IN packet through the network. The Consumers spread the IN packet broadcasting it, which updates further Consumers. This approach anticipates less Interest loss and a higher delivery ratio than NDN.

4.1 PDCN (Producer Data Controlled Notification)

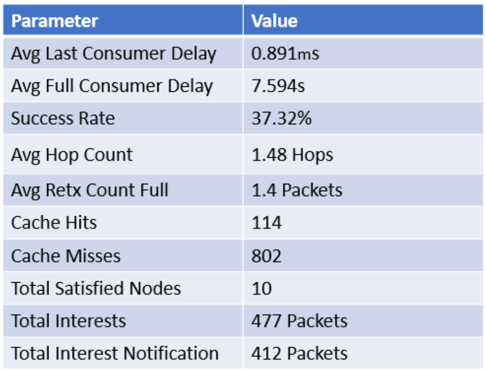
This section presents and analyses the impact of Producer mobility in NDN and PDCN.

First, it studies the results provided by the simulation with the parameters and values from table 4.1. Next, it will inspect the results of the metrics described in the proposition. Finally, it will explore the meaning and impact of each parameter value and draw some preliminary conclusions. The following tables show the first results of Producer mobility in NDN and PDCN:



Parameter	Value
Avg Last Consumer Delay	1.54ms
Avg Full Consumer Delay	31.506s
Success Rate	8.46%
Avg Hop Count	1.28 Hops
Avg Retx Count Full	1.86 Packets
Cache Hits	82
Cache Misses	1028
Total Satisfied Nodes	4
Total Interests	662 Packets

Table 4.2: NDN Metrics



Parameter	Value
Avg Last Consumer Delay	0.891ms
Avg Full Consumer Delay	7.594s
Success Rate	37.32%
Avg Hop Count	1.48 Hops
Avg Retx Count Full	1.4 Packets
Cache Hits	114
Cache Misses	802
Total Satisfied Nodes	10
Total Interests	477 Packets
Total Interest Notification	412 Packets

Table 4.3: PDCN Metrics

The tables 4.2 and 4.3 show full Consumer delay, the delay between the first Interest sent and the received Data packet, which was 4 times higher in NDN than in PDCN. The transmission time is higher in NDN because of the increased Interest retransmission, which PDCN resolves. PDCNs with FIB updated information grant the nodes the path to directly request packets avoiding broadcast.

The tables 4.2 and 4.3 reveal that PDCN has a 4.4 times higher success rate than NDN. This increase in success rate is expected since FIB maps information names to the output interface(s) that are used to forward Interest messages to the Producer. On the other hand, without FIB information, NDN broadcasts Interests hoping that a Producer receives the Interest packet. Furthermore, in PDCN, if a Consumer receives an IN with no stored and desired information, then the Consumer sends an Interest packet requesting the Data. This choice allows for a quicker Data reception which is essential for IoV dynamic topologies and non-uniform networks. The Consumer's quick response also diminishes delay constraints for demanding IoV applications. The number of Interests sent decreases from 662 packets in NDN to 477 packets in PDCN. Since FIB information is updated, the nodes do not broadcast Interest packets to the Producer.

Figures 4.1 and 4.2 show the difference between Interests and Timeout Interests (Interests that do not receive a Data response in 50ms after transmission) collected throughout the simulation. The NDN averages two Interest packets per second while PDCN has a higher Interest rate for the first 80 seconds, then decreases gradually. NDN Interest Timeout values also reach higher values than for PDCN.

Node satisfaction increases in PDCN. During the simulation, different nodes interact and start forming different clusters. The total number of nodes satisfied with Data increases from 10% to 25%. While still, 75% of the total number of nodes does not have Data, which is undesirable, further simulation ex-

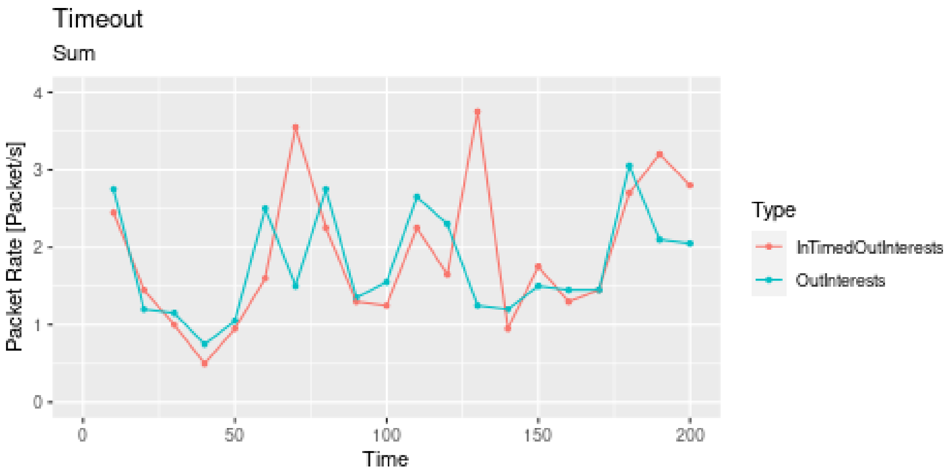


Figure 4.1: NDN Average Timeout Packet/s

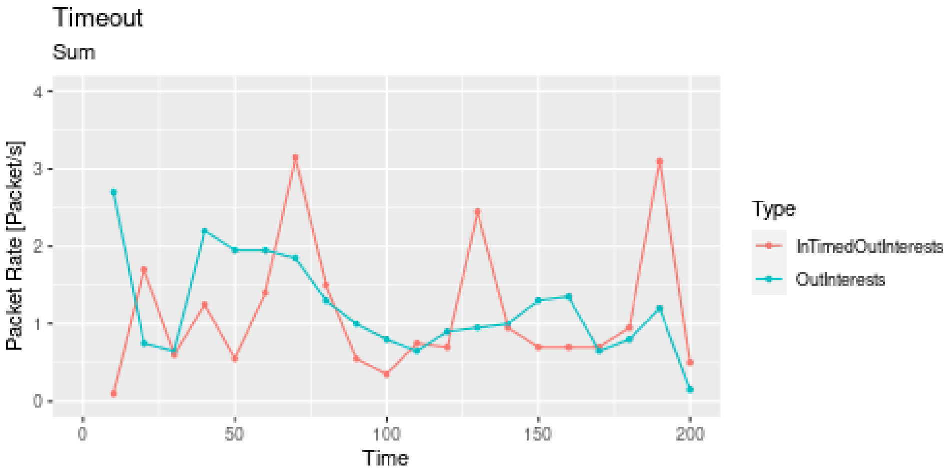


Figure 4.2: PDCN Average Timeout Packet/s

amination showed no more significant interaction between node clusters than around the 60-70 second simulation time. Around the 60-70 seconds mark, there is a peak in IN and Data packets since it is around this time that two different node clusters interact. Figures 4.5 and 4.6 show the NDN and PDCN average packets in kilobits per second from Interest and Data packets sent from all nodes presented in the simulation.

Figure 4.3 shows clearly two different node clusters, one cluster near the road intersection on the left and another smaller cluster near the road intersection on the right. The node cluster on the left encompassed nodes with Data, including the Producer. Figure 4.4 shows the moment where the clusters interacted. The node between the two clusters is in the transmission and reception range. Although the cluster interaction is present in both simulations, NDN fails to grant the same Data dissemination as PDCN.

While in NDN, these short interactions did not benefit information gathering, which helped PDCN stand out. A metric in the tables 4.2 and 4.3 which corroborates the PDCN successful information sharing is the average hop count. PDCN has a greater average hop count than NDN since FIB update broadcast increased the number of nodes with a path to the Data nodes. During IN broadcast, a path is created from the nodes receiving the packet to the Producer of that packet.

Regarding caching, PDCN cache hits are higher than NDN, and PDCN has lower cache misses. After receiving Interest packets, the node inspects his CS first, following the NDN architecture flow 2.4.

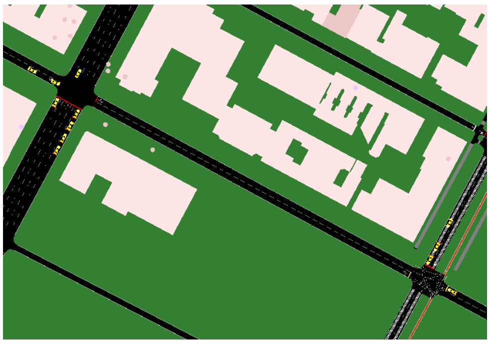


Figure 4.3: Simulation time 60 seconds

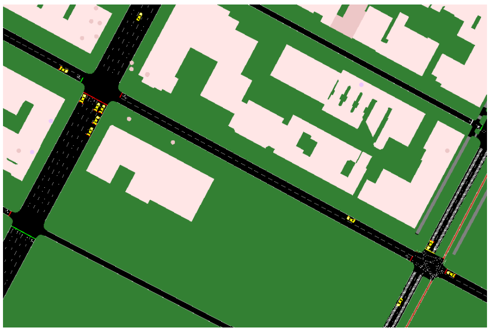


Figure 4.4: Simulation time 68 seconds

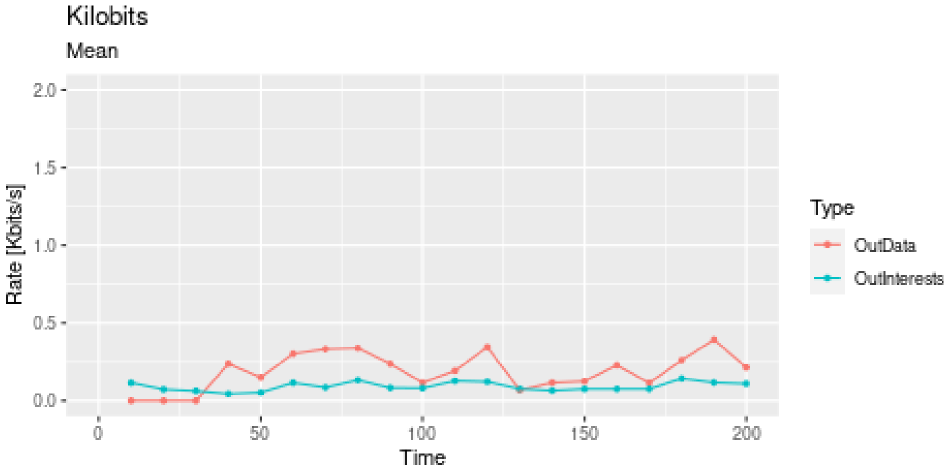


Figure 4.5: NDN Average Packets in kilobits/s

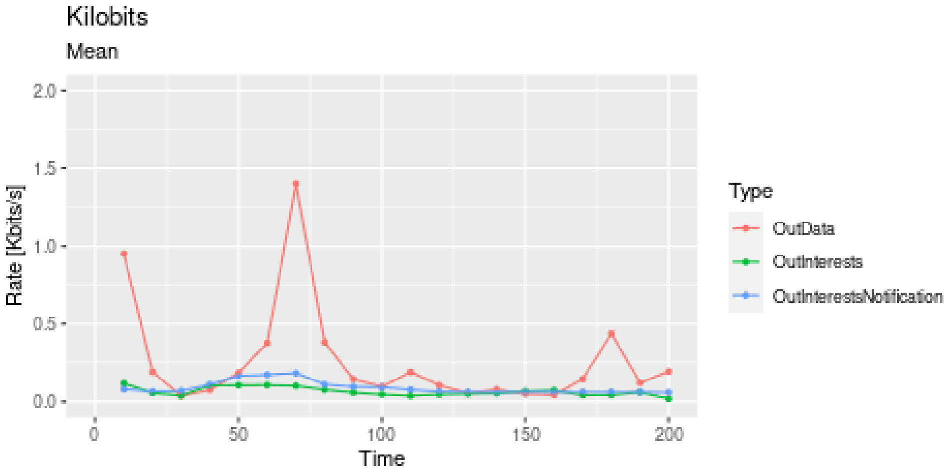


Figure 4.6: PDCN Average Packets in kilobits/s

If CS has the content cached, a cache hit occurs. Otherwise, the node emits a cache miss and will request the content depending on its PIT and FIB information. In this first results scenario, as indicated in the simulation table 4.1, the nodes request the content at a fixed rate. Nodes with content cached will receive a cache hit every time they request the content. Since PDCN has more nodes than NDN with Data cached, thanks to IN broadcast and consequent Interest success in finding the Producer and retrieving Data, the number of cache hits is higher, and the cache misses are fewer.

The number of control packets IN is 412, which is high, making the total number of packets sent from PDCN higher than NDN by 227 packets. The number of control packets is higher due to the high Producer IN frequency. A new test will be made to discover if it is possible to reach good results with lower Producer IN frequency.

The number of control packets in PDCN is 34% higher than in NDN, which is the price to pay to increase the success rate by 4.4 times and reduce the average full Consumer delay by four times in PDCN compared to NDN.

4.2 PDCN Problems and challenges

In the NDN architecture flowchart 2.4, the decision to discard or transmit the Interest depends on the FIB having a matching Interest. FIB, as explained in the NDN chapter, has two primary roles: On the one hand, the node has potential sources of matching Data to forward his Interests. FIB is the principal source of routing information. On the other hand, FIB information conceptually is not only the routing path to the required Data. It represents the node's desire to have Data matching the namespace present in FIB. To have a FIB updated is the node showing his desire for the content in FIB's namespace.

FIB updated information is critical to understanding PDCN since NDN and ndnSIM pre-route every Consumer node in the network with FIB updated information by default, which is the necessary information to request content. Without pre-routing information, Consumer nodes discard the Interest packets. For this reason, NDN does not broadcast Interests without pre-routing information because it has no FIB next-hop information to request content.

Thus, the NDN architecture was changed for nodes to broadcast Interest packets if they do not have FIB information. During the broadcast, if the Producer received the Interest, it would send the corresponding Data packet back. The Consumer would then update his FIB and cache the content. In short, the NDN architecture was changed to work in IoV scenarios, converting the Interest decision flow to work without pre-routing information.

Pre-routing and routing actualization is still a challenge in NDN architecture [33] [26].

However, PDCN is a Producer but also a routing mobility solution. In scenarios where pre-routing information is not predefined, like in IoV environments, PDCN solves the routing problem by Updating FIBs with fresh next-hop relay nodes with a Data source and by providing this information in a controlled manner through the network.

4.3 Degree Method Analysis

The First results taken show the direct result of PDCN without the degree method explained in the work proposal. This section intends to expose the differences between both implementations and explore their impacts. The PDCN with degree Method (PDCNwDM) terminology was chosen to define PDCN working with the degree method as a broadcast control decision. This terminology is only used inside this section to differentiate the method with and without the degree method. When PDCN terminology is used outside this section, it refers to PDCN with the degree method.

4.3.1 Degree Method Implementation

Two implementation decisions were made regarding the degree method broadcast conditions. The First condition is that if the IN packet receiver is the first packet Consumer, the Consumer will broadcast an IN packet. Lastly, if the Consumer receiving the IN packet has more Neighbours than the previous

Consumer node, the receiving node will broadcast the IN packet. The purpose behind this method is to avoid a control packet broadcast storm while ensuring a contained IN spread.

The following parameters were used to test PDCN with degree method (PDCNwDM):

- Neighbour list reset time:

  In an IoV scenario, nodes have a large number of interactions. In PDCNwDM, Consumer nodes save the number of interactions in a list and have a timer to refresh the list. The timer is responsible for keeping the number of neighbour nodes refreshed and accurate to the current neighbour nodes.

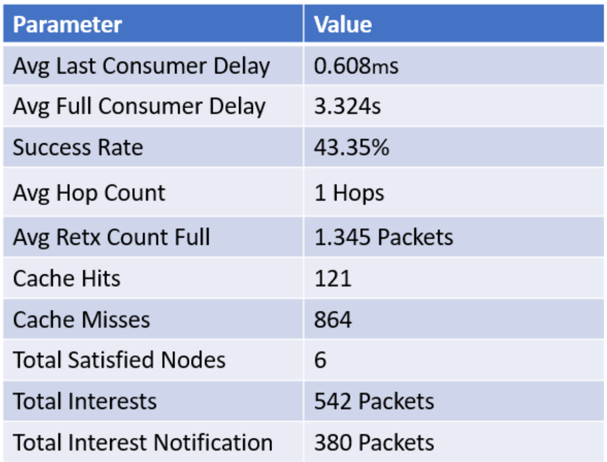
- Broadcast decision:

  The Broadcast decision values are 'high' and 'high and equal'. The Broadcast decision 'high' exemplifies the receiving Consumer node's decision to broadcast IN packets if it has more neighbour nodes than the node that sent the IN. The decision 'high and equal' represents the decision to broadcast IN packets if the node has more or the same number of neighbours as the node which sent the IN.

4.3.2 PDCN with Degree Method Results

We tested the effects of the degree method in PDCN by changing and testing these two different factors and comparing the results with PDCN without the degree method, table 4.3. In PDCNwDM, we initially set the timer to 50 seconds and the broadcast decision to 'high'.

The tests presented use the PDCNwDM metrics results from the simulation Table 4.4 as a comparison base.



Parameter	Value
Avg Last Consumer Delay	0.608ms
Avg Full Consumer Delay	3.324s
Success Rate	43.35%
Avg Hop Count	1 Hops
Avg Retx Count Full	1.345 Packets
Cache Hits	121
Cache Misses	864
Total Satisfied Nodes	6
Total Interests	542 Packets
Total Interest Notification	380 Packets

Table 4.4: PDCNwDM with Broadcast decision 'high' and 50s timer

As shown in Figure 4.7, the degree method decreases the number of IN packets by 7% compared to PDCN without the method, table 4.3. This decrease has two significant consequences:

The number of nodes receiving information drops 40%, which increases the number of cache misses seen in the table 4.4, and the number of Interests sent increases by 14%, figure 4.7. While both consequences directly result from the degree method implementation, they have different impacts.

Intending to secure a controlled broadcast of IN packets, we left 40% fewer nodes with Data by decreasing 7% in IN packets. This slight decrease in IN packets demonstrates the impact of information propagation of PDCN.

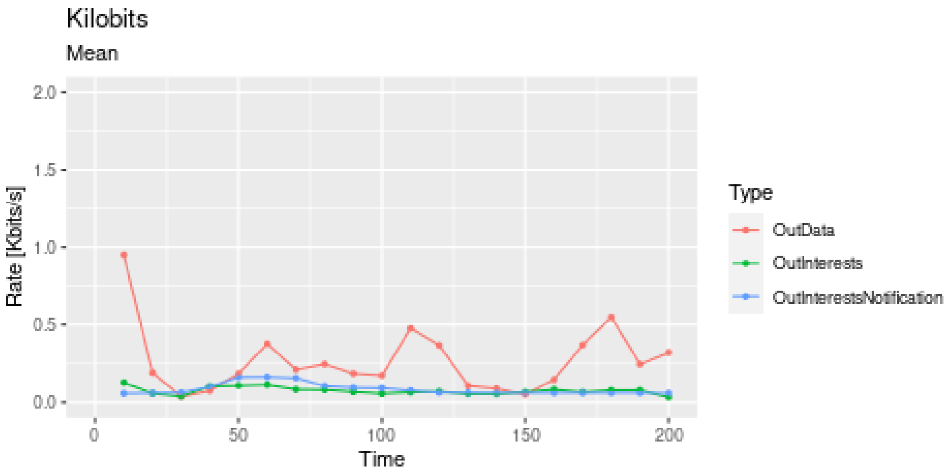


Figure 4.7: PDCNwDM Average Packets in kilobits/s

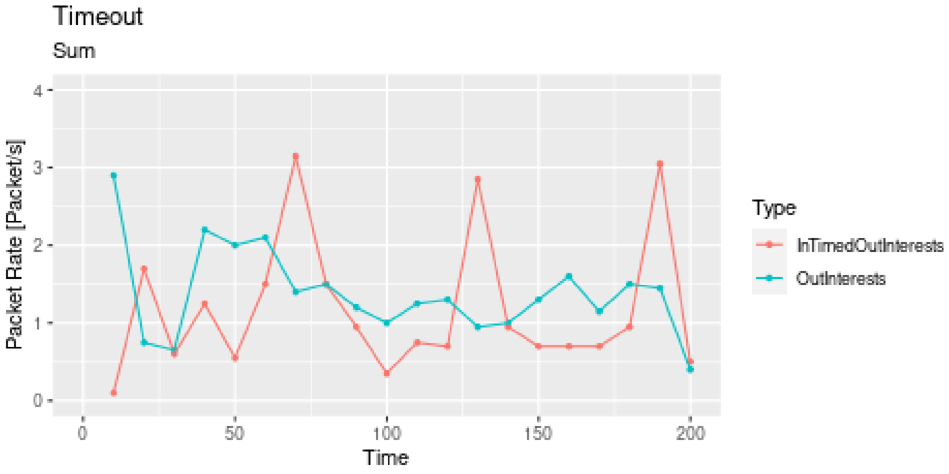


Figure 4.8: PDCNwDM Average Timeout Packet/s

However, the increase in Interest packets leads to an increase in success rate by 10% in PDCNwDM compared to PDCN, table 4.3. Furthermore, the average full delay drops by half and the average last delay decreases by 32% in PDCNwDM, table 4.4, compared to PDCN, table 4.3.

Although the increase in Interest packets benefits this approach by reducing the delays and increasing the success rate, it is at the expense of 40% of nodes not receiving any Data. Furthermore, the additional number of Interests sent are those of the nodes close to the Producer. Since there is a reduction of IN packets sent by the Producer, the network is available for the Consumers to request Data.

For the reasons stated above, a compromise must be made to reduce the delay between Consumers and Producers or increase the number of Data nodes.

For the first case, reducing the delay between Consumers and Producers depends on the type of information requested. For multimedia applications, stream videos, and applications with delay constraints, PDCNwDM would be a better choice.

Applications responsible for broadcasting higher priority information, such as car accident information, rapidly throughout the network would be of greater importance than ensuring the streaming video quality of other applications. PDCN will be better for these higher-priority information applications since the critical information would reach more nodes.

The number of timeout Interests increased 3% compared to PDCN, figure 4.8. This increase in lost Interests comes from the four nodes that could not reach nodes with Data due to the lack of FIB updated information provided by the IN packets as seen in PDCN.

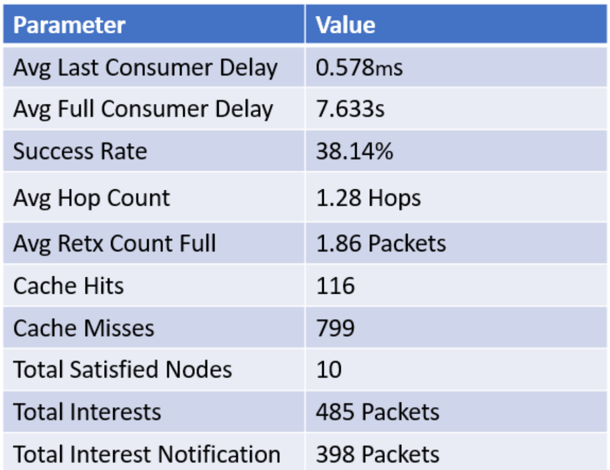
Compared to NDN, table 4.2, we see PDCNwDM, table 4.4, having a 33% increase in satisfied nodes, an Interest success rate 4.5 times higher, a shorter average last Consumer delay by 62% and a shorter full Consumer delay by 75%. The success of PDCNwDM, table 4.4 is at the expense of an increase of control packets of 33% compared to NDN, table 4.3.

After studying PDCNwDM and comparing it to NDN and PDCN, PDCNwDM has an IN-controlled broadcast contrary to PDCN, and it has a higher success rate and shorter delay than NDN, thereby it is chosen as the primary method for the tests presented in the sections below.

4.3.3 PDCNwDM Tests and Discussion

Further tests were made to PDCNwDM, changing the broadcast decision to 'high and equal' and the Neighbour list reset time.

The objective behind these tests was to improve PDCNwDM in two aspects. On the one hand, increase the number of nodes receiving Data while avoiding broadcast problems for highly dense networks. On the other hand, increase the Interest success rate and reduce Consumer delay.



Parameter	Value
Avg Last Consumer Delay	0.578ms
Avg Full Consumer Delay	7.633s
Success Rate	38.14%
Avg Hop Count	1.28 Hops
Avg Retx Count Full	1.86 Packets
Cache Hits	116
Cache Misses	799
Total Satisfied Nodes	10
Total Interests	485 Packets
Total Interest Notification	398 Packets

Table 4.5: PDCNwDM with Broadcast decision 'high and equal' and 50s timer

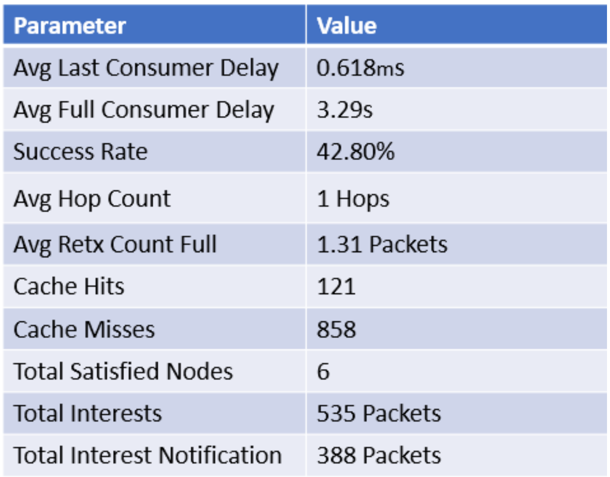
Table 4.5 reflects an increase of 2% in Interest packets and a decrease in 4% in IN packets compared to PDCN, table 4.3, but it has a decrease in 11% in Interest packets and an increase by 5% of IN packets compared to PDCNwDM, table 4.4, with 'high' as broadcast decision.

These results appoint the test with 'high and equal' to be very close to the results provided by PDCN but with the benefit of decreasing control packets, which ensured a slightly better Interest success rate and shorter Consumer delay. However, this controlled broadcast is not as restrictive as PDCNwDM, which could lead to broadcast storm problems in a different and more dense IoV scenario.

For the tests with reduced list rest time, we selected the neighbour list reset time to 10 seconds and used 'high' as the broadcast decision, table 4.6. We are strictly comparing the timer's impact on the method's performance. Consequently, we will compare it to PDCNwDM, which has 'high' and 50 seconds as implementation choices.

The results show a slight decrease in success rate by 1% and an increase in both Consumer delays by 1% and 1% for last and full delays, respectively. The results exhibit the same pattern as verified in other performance comparisons. Reducing Interest packets and increasing IN packets lead to a decrease in Interest success rate.

These results imply that frequently resetting the list of neighbours to zero does not strongly impact the



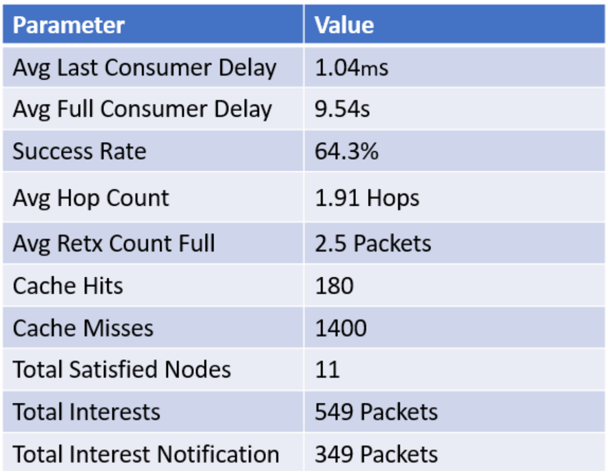
Parameter	Value
Avg Last Consumer Delay	0.618ms
Avg Full Consumer Delay	3.29s
Success Rate	42.80%
Avg Hop Count	1 Hops
Avg Retx Count Full	1.31 Packets
Cache Hits	121
Cache Misses	858
Total Satisfied Nodes	6
Total Interests	535 Packets
Total Interest Notification	388 Packets

Table 4.6: PDCNwDM with Broadcast decision 'high' and 10s timer

Interest and IN rate, which consequently translates to a lower impact on the success rate and Consumer delay.

4.4 PDCNwDM with Zipf Distribution

In this section, we tested Consumers requesting content following the Zipf–Mandelbrot distribution (with the parameter of power and improvement rank as 0.7). The number of content selected was 0.026I/s, table 4.1. Finally, we selected the Broadcast decision to 'high' and 50 seconds as reset neighbour list timer. We choose PDCNwDM as the base architecture. Figure 4.10, shows the Zipf impact on Interest timeout which decreased 36% compared to PDCNWDM, figure 4.8. Table 4.7 shows the results of the tests made to Consumers with Zipf-Mandelbrot.



Parameter	Value
Avg Last Consumer Delay	1.04ms
Avg Full Consumer Delay	9.54s
Success Rate	64.3%
Avg Hop Count	1.91 Hops
Avg Retx Count Full	2.5 Packets
Cache Hits	180
Cache Misses	1400
Total Satisfied Nodes	11
Total Interests	549 Packets
Total Interest Notification	349 Packets

Table 4.7: PDCNwDM with Zipf-Mandelbrout

PDCNwDM with Zipf-Mandelbrot, table 4.7 has 48% higher Interest success rate than PDCNwDM, table 4.4. Furthermore, PDCNwDM with Zipf-Mandelbrot, table 4.7, number of Interest packets sent increased by 1%, and the number of control packets sent decreased by 8% compared to PDCNwDM, table 4.4.

When analysing the difference between PDCN, table 4.3, and PDCNwDM, table 4.4, the results suggest that the decrease in IN packets increased the success rate by 10%, while with Zipf, table 4.7,

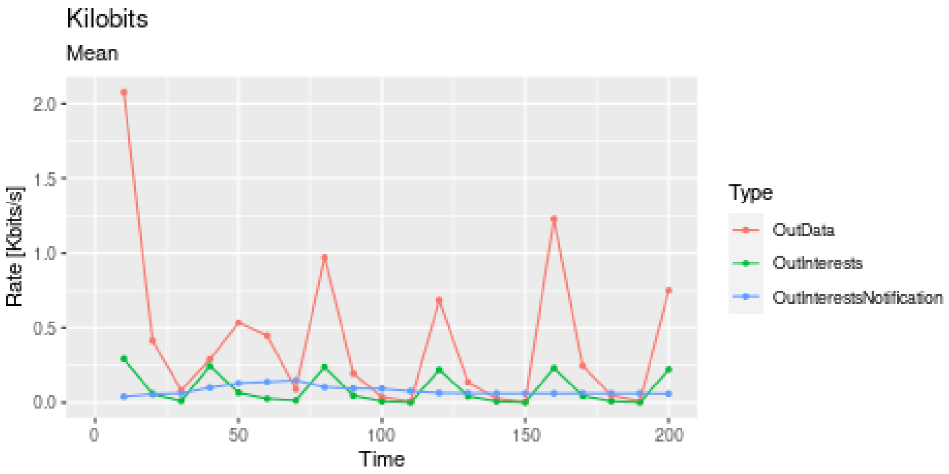


Figure 4.9: PDCNwDM with Zipf Average Packets in kilobits/s

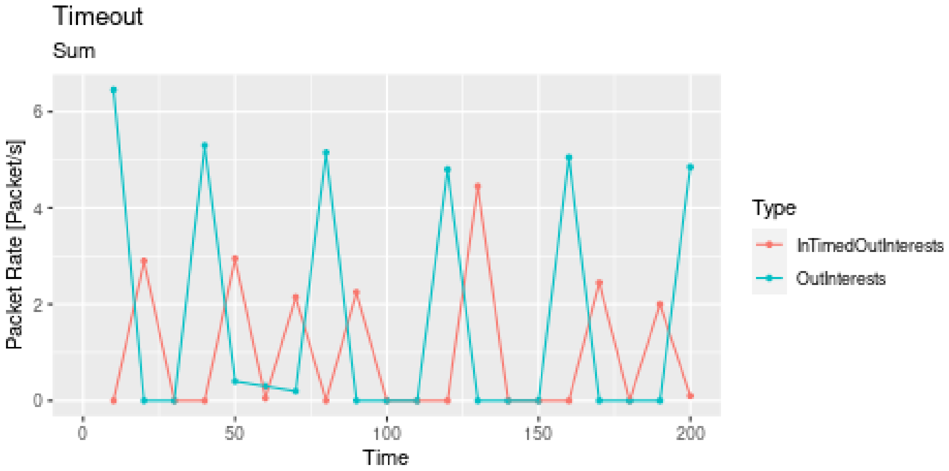


Figure 4.10: PDCNwDM with Zipf Average Timeout Packet/s

the increase was 48% . In that comparison, the number of nodes receiving Data decreased by 40% while for PDCNwDM with Zipf, it increased by 83%.

These results come from the fact that the Zipf-Mandelbrot distribution pattern reduces the most significant packet drop probability compared to the use of random distribution [65]. In PDCNwDM with Zipf-mandelbrot distribution, the transmission on the link between nodes is not filled by Data traffic and Interest packets when compared to random distribution patterns, figure 4.9.

Figure 4.10, shows the Zipf impact on Interest timeout which decreased 36% compared to PDCNWDM, figure 4.8.

4.5 IoV Density, Consumer and Producer Impact

In this section, we study the node density impact on PDCNwDM and NDN. We start by studying the Consumer node's density variation impact on both architectures. We intend to analyse the number of Consumers and Producers' impact on NDN and PDCNwDM before scaling to different and complex IoV simulations.

In order to study Consumer impact, we selected three different numbers of Consumers. We selected 40 vehicles as the base scenario, 39 Consumers and one Producer, then selected 120 for the second level and finished on 160 nodes. With each increasing level, the simulation took more time to finish, which had two reasons:

More Consumer and Consumer interaction increases the number of Interest broadcasts and, consequently, the number of Interest packets throughout the network. This increase in the number of Consumer nodes is not a problem for PDCNwDM, which avoids Interest broadcasts using IN. However, NDN has only Interest broadcast as a mechanism to reach and find Producers.

Secondly, we made these tests using the Manhattan model, on 1400m x 1400m map. The number of junctions and streetlights encourages node aggregation. Furthermore, the 1400m x 1400m is a small space for increased node density which can be seen in Figure 4.11.

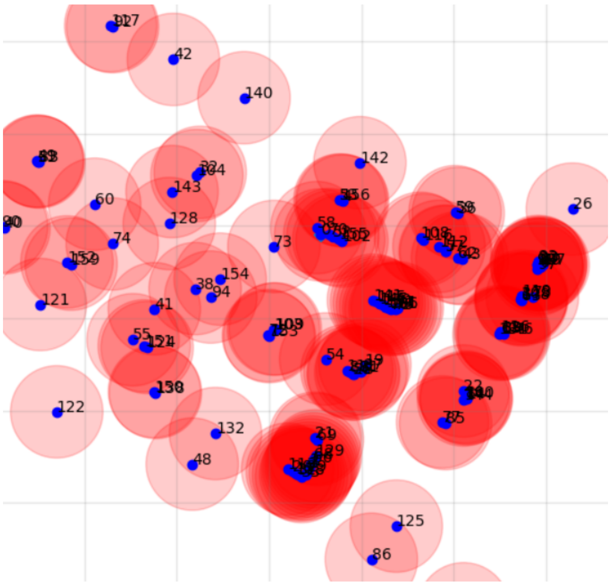


Figure 4.11: Simulation density with 160 nodes in a 1400m x 1400m map.

Figure 4.11 shows vehicles by dots and their communication range by red circles. The numbers above the dots represent node ids. The overlapping node's communication range increases the red circle's opacity.

In level 3, 160 nodes, the NDN simulation stopped running, and the increasing number of broadcast storms led to the simulation failure. However, PDCNwDM ran with 160 nodes, and we show and analyse the results.

After Consumer variation analysis, we incorporate more Producers into the network and study their impact on PDCNwDM and NDN.

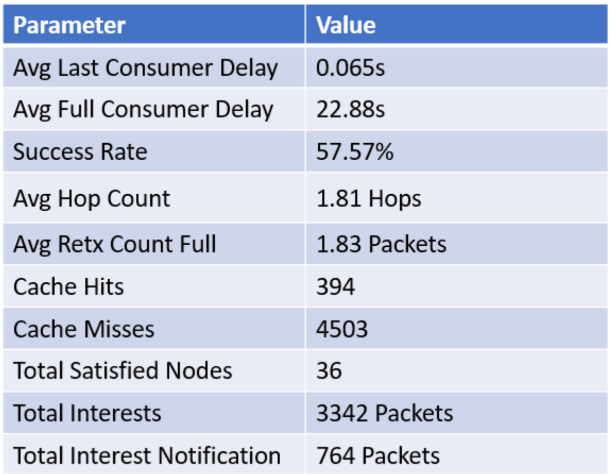
We tested the values of PDCNwDM and NDN using the values presented in the table 4.1 except the number of nodes that we changed according to the abovementioned levels. In the PDCNwDM method, We selected 'high' as the broadcast decision and 50 seconds for the reset neighbour list timer.

4.5.1 IoV Consumer Impact

We start by comparing PDCNwDM with 120 nodes, as shown in table 4.8 against PDCNwDM with 40 nodes, table 4.4. In the Consumer delay 120 nodes scenario, the average last delay is 107 times higher than for the 40 nodes scenario, and the average full delay is seven times higher.

Increasing the number of Consumers and the discrepancy between Consumers and Producers leads to the increased values of:

- Number of Interests by 517%;



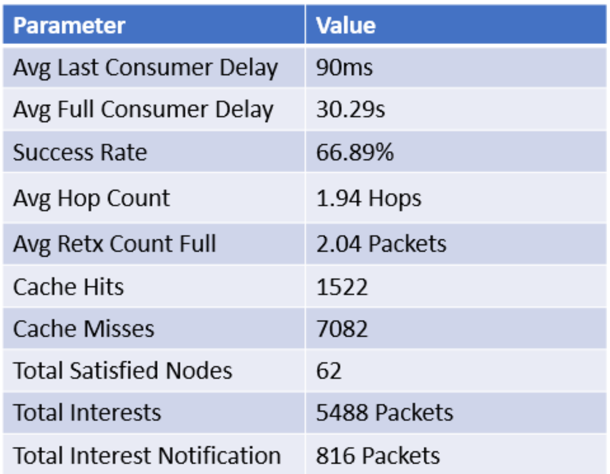
Parameter	Value
Avg Last Consumer Delay	0.065s
Avg Full Consumer Delay	22.88s
Success Rate	57.57%
Avg Hop Count	1.81 Hops
Avg Retx Count Full	1.83 Packets
Cache Hits	394
Cache Misses	4503
Total Satisfied Nodes	36
Total Interests	3342 Packets
Total Interest Notification	764 Packets

Table 4.8: PDCNwDM 120 nodes and 1 Producer

- Total number of packets by 345%;
- Average packet re-transmission by 36%;
- Average hop count increased by 81%;
- Average Full Consumer delay by 595%;

Compared with PDCNwDM with 40 nodes, table 4.4.

Within a vehicle-dense environment, the interactions between nodes increase, increasing the number of broadcast INs. Adding to the increased number of Interests packets leads to an increased number of satisfied nodes and Interest success rate (increased by 33%) While in PDCNwDM with 40 nodes, table 4.4, the Data could only reach 15% of the nodes, with 120 nodes, table 4.8, PDCNwDM reaches 29% of the total nodes. Although the increase in Consumer delay would be, at first sight, a signal of worst performance, it is adequate for the increased distance and hops that the packet needs to take to reach the Consumer. Finally, we conclude that for 120 nodes and 119 Consumers, table 4.8, PDCNwDM has a 34% increase in Interest success rate and a 500% increase in node satisfaction for a total packet increase of 345%, compared with PDCNwDM with 40 nodes, table 4.4.



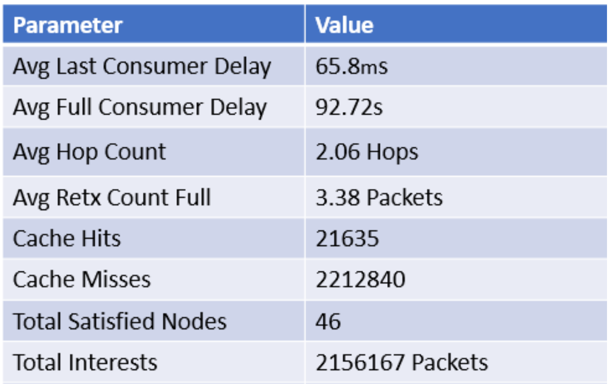
Parameter	Value
Avg Last Consumer Delay	90ms
Avg Full Consumer Delay	30.29s
Success Rate	66.89%
Avg Hop Count	1.94 Hops
Avg Retx Count Full	2.04 Packets
Cache Hits	1522
Cache Misses	7082
Total Satisfied Nodes	62
Total Interests	5488 Packets
Total Interest Notification	816 Packets

Table 4.9: PDCNwDM 160 nodes and 1 Producer

Table 4.9, shows the results of PDCNwDM with 1 Producer and 159 Consumers.

As shown in Figure 4.11, the increased density of the network nodes becomes more accessible for nodes to get Data from different clusters since there are more interactions. However, with increasing Consumers, more Interest packets are sent. Fortunately, IN packets and, consequently, FIB updates reduce Interest broadcasts. The number of satisfied nodes increased by 34% from the previous simulation.

The average Consumer delays, Hop counts and re-transmissions have a slight value increase as expected.



Parameter	Value
Avg Last Consumer Delay	65.8ms
Avg Full Consumer Delay	92.72s
Avg Hop Count	2.06 Hops
Avg Retx Count Full	3.38 Packets
Cache Hits	21635
Cache Misses	2212840
Total Satisfied Nodes	46
Total Interests	2156167 Packets

Table 4.10: NDN 120 nodes and 1 Producers

A high vehicle-dense scenario like this and NDN's lack of broadcast control lead to broadcast storms. Unlike PDDCNwDM, NDN has to use Interest broadcast to reach for Data, table 4.10 exhibits an increase in Interest packets by 64417% compared to PDCNwDM. Broadcast storms lead to severe contention at the link layer, contention delay, packet collisions, inefficient bandwidth use, and more [66]. On Consumer delay, the average full delay increases by 305% compared to PDCNwDM, table 4.8. On total satisfied nodes, only 39% receive Data packets.

During the analysis, we could not get the Interest success rate to compare with other mechanisms. The Timeout Event did not send events according to the number of Interest lost. It is with this event that we calculate the Interest success rate. We assume that the timeout event relates somewhat to the Interest broadcast storm problem. Further analysis will be made to address this issue.

However, we can still assume that NDN was unsuccessful because it failed to satisfy most of the network nodes, only 39% of the total nodes present were satisfied. The number of Interest packets sent explicitly shows the presence of a broadcast storm.

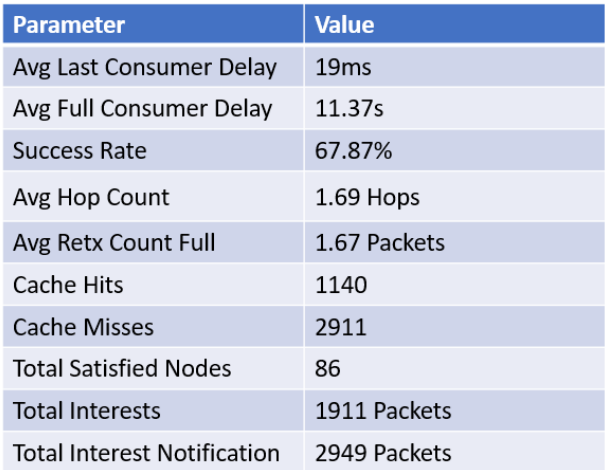
Compared to PDCNwDM with 120 nodes, table 4.8, NDN with 120 nodes, table 4.10, satisfied 28% more Consumers but at the cost of uncontrolled Interest broadcast, making NDN unreliable for this number of Consumers.

Finally, for PDCNwDM increasing the number of Consumers had an increase in success rate as well as total nodes satisfied through the three load levels, which implied that the increase in Consumer nodes tests had a positive impact on PDCNwDM. For NDN, the increase in Consumer nodes led to uncontrolled broadcast, which was expected, solidifying NDN as unreliable in dense Consumer node environments.

4.5.2 IoV Producer Impact

In this section, we compare PDCNwDM with six Producers to 1 Producer to study the Producers' impact on the simulation, table 4.11. Moreover, we compare PDCNwDM and NDN with 120 nodes, table 4.12 and finally correlate the performance of both architectures with 40 and 120 nodes.

Adding Producers to the scheme increased the number of IN packets by 286% compared to PDCNwDM with one Producer, table 4.8, and raised Data availability on the network.



Parameter	Value
Avg Last Consumer Delay	19ms
Avg Full Consumer Delay	11.37s
Success Rate	67.87%
Avg Hop Count	1.69 Hops
Avg Retx Count Full	1.67 Packets
Cache Hits	1140
Cache Misses	2911
Total Satisfied Nodes	86
Total Interests	1911 Packets
Total Interest Notification	2949 Packets

Table 4.11: PDCNwDM 120 nodes and 6 Producers

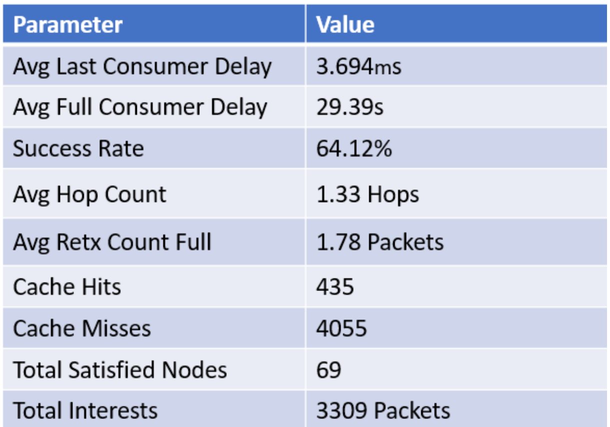
As a consequence of IN packets increase, we obtained the following metrics increase/decrease compared to PDCNwDM with one Producer, table 4.8:

- Average Last Consumer delay decreased by 70%;
- Average Consumer delay decreased by 50%;
- Average packet re-transmission decreased by 9%;

The PDCNwDM with 6 producers, table 4.11, increase in Data availability and an increase in total satisfied nodes diminishes the number of hops that an Interest has to make to reach content, which explains the average hop count decreased by 6.6%, compared to PDCNwDM with one producer, table 4.8. Although the number of total packets increased by 18%, adding five Producers reduced Consumer delays, packet re-transmission, increased Total satisfied nodes by 139% and Interest Success rate by 18%, compared to PDCNwDM with one producer, table 4.8.

These results suggest that PDCNwDM has a more positive impact in networks with less Producers. We can also expect that keeping adding Producers to the network would lead to more IN control packets than Interests and eventually lead to PDCNwDM decreased performance. Since PDCNwDM success is not based on IN directly but on Interest packets routing success.

Table 4.12 shows the impact that the number of Producers have in NDN.



Parameter	Value
Avg Last Consumer Delay	3.694ms
Avg Full Consumer Delay	29.39s
Success Rate	64.12%
Avg Hop Count	1.33 Hops
Avg Retx Count Full	1.78 Packets
Cache Hits	435
Cache Misses	4055
Total Satisfied Nodes	69
Total Interests	3309 Packets

Table 4.12: NDN 120 nodes and 6 Producers

Compared to PDCNwDM, table 4.11, NDN, table 4.12, has 159% higher total Consumer delay;

the success rate is 6% lower and 47% less total packets. These results suggest that NDN performs worse in delay constraint applications for this scenario than PDCNwDM. The price to pay for decreasing Consumer delay and increasing Satisfied Nodes is to increase the number of control packets by 47% compared to NDN, table4.12.

Although NDN performance is still worse than PDCNwDM, it is closer, which can be seen by the Interest success rate. Furthermore, adding five more Producers to NDN avoided the broadcast storm analysed in the previous section.

We can conclude that Producer variation significantly impacts NDN and PDCN. For NDN, table 4.12, it avoids broadcast storms and is responsible for a decrease in Consumer delay by 68% and total node content satisfaction by 50% compared to NDN with one Producer, table 4.10. For PDCNwDM, table 4.11, it showed a decrease in full Consumer delay by 55% and an increase in total satisfied nodes by 138.8% compared to PDCNwDM with one Producer, table 4.8.

4.6 Final Discussion and Analysis

This Thesis aims to solve the Producers' mobility problem framed in the IoV environment. We divide the discussion on the IoV and Producer mobility problem and tackle the challenges brought by both dimensions. The IoV environment is characterized as a large-scale network due to the number of interactions between diverse elements in a complex network [25]. IoV is a network whose topology is constantly changing due to the movement, and alternating speed of the different nodes in the network [25]. The topology change also generates different internal networks with different densities. These characteristics cause difficulties in operation to ITS and IoV applications such as vehicular safety applications, road traffic efficiency and infotainment applications, and Video streaming [1] [24].

IoV applications require low service delay and reliable network communication, i.e., real-time communication, even when communication or network nodes malfunction. Our proposal solves the problem derived from the characteristics of IoV. The results from the different tests on Producer mobility, PDCN always managed to reduce Consumer delay compared to NDN. The maximum Consumer delay reduction compared to NDN was 848%, table 4.4 compared to NDN, table 4.2. The maximum Consumer delay was in an IoV environment with 40 nodes, where 39 were Consumers, and one was the Producer. Therefore, PDCN has a full Consumer delay reduction for environments with low node density compared to NDN.

Where for high density IoV scenarios, PDCN Consumer delay, table 4.8 is 305% lower than NDN, table 4.10. However, the results on Consumer delay also depend on the percentage of Consumer and Producer nodes per Total nodes. The increase in Producers decreases the Consumer delay by 50% from 1 Producer table 4.8, to 6 Producers, table 4.11, in a 120 node scenario.

Moreover, PDCN is more reliable than NDN as it also assures a more significant number of satisfied nodes in lower node densities. For a low density scenario (40 nodes), PDCN, table 4.4, had 50% more nodes satisfied than NDN, table 4.2. For a high density scenario (120 nodes, 1 Producer and 119 Consumers) NDN, table 4.10 has a higher node satisfaction rate than PDCN, table 4.8 by 28%. However, in this scenario, Consumer density greatly impacts the node satisfaction rate since NDN creates a broadcast storm. The broadcast storm, established by the number of Interests sent 4.10, makes it unreliable to consider NDN to have a better node satisfaction rate in high node dense scenarios than PDCN.

For a high density scenario having 120 nodes with 6 Producers, the node satisfaction rate in PDCN, table 4.11 is 25% higher than NDN, table 4.12. This result proves that PDCN has a larger node satisfaction rate than NDN in a higher Producer and node density scenario.

The challenge arising from the Producers mobility problem in NDN that we intend to solve is the loss and discard of Interest packets during producer movement. Producer mobility creates Invalid FIB entries, useless PIT entries, request delivery failure and increased handover latency. Consumers without up-to-date routing tables send Interest packets following the old path of the Producer, taking up bandwidth and eventually getting lost by not reaching the Producer [56] [57] [50]. These aimless Interest packets on the network increase the content response delay from Producers to Consumers.

We solve the Producer mobility problem by adding a new IN packet to the Producer that the Producer proactively sends to the network. The Producer with the IN packet establishes an updated path through the nodes that receive this packet. Nodes update FIB tables according to the received IN packet. Consumer Interest packets directed to the content provided by the Producer will travel through the intermediate nodes or directly to the Producer, guaranteeing request delivery success, as we can analyse by the various tests produced.

On the baseline scenario (40 nodes, low node density), PDCN, table 4.4 verifies an increase in Interest success rate by 412% compared to NDN, table 4.2. For the 120 nodes and one Producer scenario, we could not measure the NDN Interest success rate due to the broadcast storm caused by the increased node density and the total Consumers per node. Broadcast storms hinder Interest success rate, efficient bandwidth usage, Consumer delay, and more [66]. PDCN avoids broadcast storms in the same scenario and the scenario with an increased number of Consumers, reinforcing the impact of IN on PDCN's success.

In a high node dense scenario, PDCN, table 4.11 has 6% higher Interest success rate than NDN, table 4.12. With high density environments, higher Data availability and node interaction, it becomes easier for NDN to reach Producers, decreasing the Interest success rate discrepancy.

In addition, the IN packet's impact on the architecture is in the Interest packet re-transmission reduction. Since the path to the Producer is updated, and the Interest packets are successfully delivered to the Consumer, the Interest re-transmission and the load/volume on the network/bandwidth diminishes. This load reduction on the network increases the network response speed, verified in the PDCN results. The re-transmission is directly related to the Interest success rate. The higher the Interest success rate, the lower the Interest re-transmission. For example on the baseline scenario, PDCN, table 4.4 has a lower average packet re-transmission rate by 38% compared to NDN, table 4.2.

Finally, the PDCN results, in general, showed that the solution is promising, as it obtained good results compared to the literature works. On the other hand, some open points exist, such as the performance in different IoV scenarios. Despite the proposed solution, further investigation is needed in this area with varying scenarios and metrics.

Chapter 5

Final Considerations

This Thesis analysed and presented a solution for Producer mobility in a realistic vehicular scenario for the ICN architecture. Firstly, the Thesis showed the growth in the complexity of vehicle systems and the demanding communication efficiency in these interconnected networks. ICN Networks were chosen as a prominent architecture to provide moving nodes with a better communication solution than IP communication. The Thesis introduced ICN Networks' advantages and disadvantages regarding moving nodes. According to the literature, Producer mobility is still an ICN problem. Thus, the Thesis focused on solving the Producer mobility problem for dynamic vehicle scenarios. The Thesis shows the research on NDN, ICN, IoV, ITS and Producer mobility. The study showed successful Consumer content transmission during its movement as the NDN architecture covered it. However, the NDN architecture provided the Producer mobility with no assistance. Research showed Producer mobility disadvantages that needed to be tackled.

5.1 Achievements

This Thesis contributes to the current Producer mobility literature, in Chapter 3 shows the scientific study behind the proposed work and introduces the PDCN routing protocol. PDCN uses Interest Notification packets to provide Consumer nodes with an updated path to the desired content information. The IN packets are sent in a controlled manner through the network, ensuring the nodes that it enters in contact with an updated FIB.

Chapter 4 features the proposed solution's different test outcomes. PDCN and NDN performance was tested against node density and Consumer and Producer density variation. The impact of the Consumer broadcast strategy type was also tested in PDCN. These tests were essential to assess the validity of PDCN as a successful routing mechanism.

PDCN showed promising results by increasing the Interest success rate by 412% and lower average re-transmission Packets by 38% compared to NDN in low node density scenarios. The success of these results solved the Producer mobility problems introduced by NDN, such as the unnecessary Interest packet losses on the old path towards the Producer. Furthermore, PDCN intended to resolve the issues created by different IoV environments and conditions, such as highly dynamic node movements and different density-like city scenarios. PDCN exhibited encouraging results in different density scenarios, reducing Consumer delay by 848% and increasing node satisfaction rate by 50% compared to NDN in low node density scenarios.

Mobility management is crucial for proper network functioning, and the Producer mobility problem needs to be resolved. This Thesis contributes to the literature on this problem showing promising results compared to the current literature.

5.2 Limitations

In future work, more than one SUMO simulation is needed to represent all the different vehicle scenarios, environments and vehicle conditions. For this reason, PDCN should be analysed against different SUMO simulations.

For future work in PDCN, further analysis should be made of PDCN in different IoV environments to test PDCN results for the different metrics. Further statistical analytics methods should be used in PDCN to retrieve more accurate results.

PDCN communication and complexity should be tested with different content. Producers and Consumers with increased content diversity should be implemented and tested to analyse its impact in an IoV environment. PDCN and NDN communication diversity and medium diversity should also be considered. For further comparison, the map-me proposal explained in the proposal section should be implemented and used as a comparison to PDCN.

Finally, further tests should be made to the namespace and handover capabilities of PDCN. Although the PDCN routing mechanism allows consumers that wish for the content to retrieve it in contact with the producer or with consumers with a FIB updated path to the Producer, Producer handover through a hierarchical namespace should also be tested and impact analysed.

5.3 Future Work

Research on Producer mobility is still recent compared to other ICN features. Further investigation is still needed to ensure Producer mobility support in all aspects. Several works and investigations can be derived from this thesis and are characterized in several areas such as: software-defined, Mobility and real-time applications, IoT and VANETs. Other issues, such as QoS/QoE-based decisions, scalability, security and privacy, are involved in almost all aspects of the areas mentioned above.

The following two works propose Interesting new ideas for ICN networks that could be integrated into PDCN as future work.

5.3.1 Fog computing based content-aware taxonomy for caching optimization in information-centric networks

Wang et al. [11] propose an efficient information-centric mechanism integrated with fog computing to reduce the network's total number of caching content by labelling the dynamic and user-shareable data.

Wang et al. try to resolve the limited memory and processor speed of modern routers with fog's content-aware ability and computation capacity. The framework presented uses fog computing as a middle level between global network ICN and underlying networks. The framework uses fog nodes and the taxonomy mechanism in ICN to classify data into user-shareable and non-shareable data before transferring it to the global network. While the fog does the computation work, routers can focus on the forwarding and maintain the benefits of in-network caching, reducing latency.

Further work on fog could provide mobility support to the content Producer, using fog's computation, storage and location-aware capabilities. Fog's capabilities could be used to design a mobility support mechanism for data exchange and communication considering the features of IoV service (e.g., traffic communication, infotainment, and more) and the IoV environment.

5.3.2 DeepNDN: Opportunistic Data Replication and Caching in Support of Vehicular Named Data

Manzo et al. [67] propose DeepNDN, a communication scheme based on the joint application of NDN and probabilistic spatial content caching, which enables content retrieval in fragmented and dynamic network topologies with tight delay constraints.

The DeepNDN scheme uses Convolutional Neural Network (CNN) architecture to capture the complex relations between Spatio-temporal patterns of mobility and content requests and DeepNDN performance.

For example, when a Producer residing in a road segment comes in contact with a node in a different road segment, the content is replicated to the node with a probability b. The node receiving the content can forward or keep the content with probability k.

The scheme requires a coordination and management function that, given a minimum hit ratio r and a maximum delay for a content, modulates (b,k) over space to satisfy a set amount of network and application requirements.

Manzo et al. rely on opportunistic communications to guarantee content persistence and tight QoS levels while optimizing resource utilization in the vehicular environment.

Bibliography

[1] N. Lu, N. Cheng, N. Zhang, X. Shen, and J. W. Mark, "Connected vehicles: Solutions and challenges," *IEEE Internet of Things Journal*, vol. 1, no. 4, pp. 289–299, 2014.

[2] J. Contreras-Castillo, S. Zeadally, and J. A. Guerrero-Ibañez, "Internet of vehicles: Architecture, protocols, and security," *IEEE Internet of Things Journal*, vol. 5, no. 5, pp. 3701–3709, 2018.

[3] S. Mastorakis, A. Afanasyev, and L. Zhang, "On the evolution of ndnsim: An open-source simulator for ndn experimentation," *SIGCOMM Comput. Commun. Rev.*, vol. 47, p. 19–33, Sept. 2017.

[4] M. Amadeo, C. Campolo, and A. Molinaro, "Information-centric networking for connected vehicles: a survey and future perspectives," *IEEE Communications Magazine*, vol. 54, no. 2, pp. 98–104, 2016.

[5] L. Zhang, A. Afanasyev, J. Burke, V. Jacobson, k. claffy, P. Crowley, C. Papadopoulos, L. Wang, and B. Zhang, "Named data networking," *SIGCOMM Comput. Commun. Rev.*, vol. 44, p. 66–73, July 2014.

[6] M. A. Hussaini, S. A. Nor, and A. A. Ahmad, "Producer mobility support schemes for named data networking: A survey," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 8, p. 5432, Jan 2018.

[7] J. Augé, G. Carofiglio, G. Grassi, L. Muscariello, G. Pau, and X. Zeng, "Map-me: Managing anchorless producer mobility in content-centric networks," *IEEE Transactions on Network and Service Management*, vol. 15, no. 2, pp. 596–610, 2018.

[8] C. Perkins, "Mobile ip," *IEEE Communications Magazine*, vol. 35, no. 5, pp. 84–99, 1997.

[9] B. Ahlgren, C. Dannewitz, C. Imbrenda, D. Kutscher, and B. Ohlman, "A survey of information-centric networking," *IEEE Communications Magazine*, vol. 50, no. 7, pp. 26–36, 2012.

[10] H. Khelifi, S. Luo, B. Nour, H. Moungla, Y. Faheem, R. Hussain, and A. Ksentini, "Named data networking in vehicular ad hoc networks: State-of-the-art and challenges," *IEEE Communications Surveys Tutorials*, vol. 22, no. 1, pp. 320–351, 2020.

[11] M. Wang, J. Wu, G. Li, J. Li, Q. Li, and S. Wang, "Toward mobility support for information-centric IoV in smart city using fog computing," in *2017 IEEE International Conference on Smart Energy Grid Engineering (SEGE)*, pp. 357–361, 2017.

[12] M. Wang, J. Wu, G. Li, J. Li, and Q. Li, "Fog computing based content-aware taxonomy for caching optimization in information-centric networks," in *2017 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, pp. 474–475, 2017.

[13] L. Figueiredo, I. Jesus, J. A. T. Machado, J. R. Ferreira, and J. L. Martins de Carvalho, "Towards the development of intelligent transportation systems," in *ITSC 2001. 2001 IEEE Intelligent Transportation Systems. Proceedings (Cat. No.01TH8585)*, pp. 1206–1211, 2001.

[14] F. Wang, "Parallel control and management for intelligent transportation systems: Concepts, architectures, and applications," *IEEE Transactions on Intelligent Transportation Systems*, vol. 11, no. 3, pp. 630–638, 2010.

[15] G. Dimitrakopoulos, "Intelligent transportation systems based on internet-connected vehicles: Fundamental research areas and challenges," in *2011 11th International Conference on ITS Telecommunications*, pp. 145–151, 2011.

[16] K. M. Alam, M. Saini, and A. E. Saddik, "Toward social internet of vehicles: Concept, architecture, and applications," *IEEE Access*, vol. 3, pp. 343–357, 2015.

[17] J. Zhang, F. Wang, K. Wang, W. Lin, X. Xu, and C. Chen, "Data-driven intelligent transportation systems: A survey," *IEEE Transactions on Intelligent Transportation Systems*, vol. 12, no. 4, pp. 1624–1639, 2011.

[18] N. R. Moloisane, R. Malekian, and D. Capeska Bogatinoska, "Wireless machine-to-machine communication for intelligent transportation systems: Internet of vehicles and vehicle to grid," in *2017 40th International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)*, pp. 411–415, 2017.

[19] S. Al-Sultan, M. M. Al-Doori, A. H. Al-Bayatti, and H. Zedan, "A comprehensive survey on vehicular ad hoc network," *Journal of Network and Computer Applications*, vol. 37, pp. 380 – 392, 2014.

[20] J. Yin, T. ElBatt, G. Yeung, B. Ryu, S. Habermas, H. Krishnan, and T. Talty, "Performance evaluation of safety applications over DSRC vehicular ad hoc networks," in *Proceedings of the 1st ACM International Workshop on Vehicular Ad Hoc Networks*, VANET '04, (New York, NY, USA), p. 1–9, Association for Computing Machinery, 2004.

[21] K. Zhang, Y. Mao, S. Leng, Y. He, and Y. ZHANG, "Mobile-edge computing for vehicular networks: A promising network paradigm with predictive off-loading," *IEEE Vehicular Technology Magazine*, vol. 12, no. 2, pp. 36–44, 2017.

[22] J. Wan, J. Liu, Z. Shao, A. Vasilakos, M. Imran, and K. Zhou, "Mobile crowd sensing for traffic prediction in internet of vehicles," *Sensors*, vol. 16, p. 88, Nov 2016.

[23] J. Contreras-Castillo, S. Zeadally, and J. A. G. Ibáñez, "A seven-layered model architecture for internet of vehicles," *Journal of Information and Telecommunication*, vol. 1, p. 4–22, Feb 2017.

[24] W. Duan, J. Gu, M. Wen, G. Zhang, Y. Ji, and S. Mumtaz, "Emerging technologies for 5g-iov networks: Applications, trends and opportunities," *IEEE Network*, vol. 34, no. 5, pp. 283–289, 2020.

[25] W. Wu, Z. Yang, and K. Li, "Chapter 16 - internet of vehicles and applications," in *Internet of Things* (R. Buyya and A. Vahid Dastjerdi, eds.), pp. 299–317, Morgan Kaufmann, 2016.

[26] G. Xylomenos, C. N. Ververidis, V. A. Siris, N. Fotiou, C. Tsilopoulos, X. Vasilakos, K. V. Katsaros, and G. C. Polyzos, "A survey of information-centric networking research," *IEEE Communications Surveys Tutorials*, vol. 16, no. 2, pp. 1024–1049, 2014.

[27] M. Amadeo, C. Campolo, and A. Molinaro, “Named data networking for priority-based content dissemination in vanets,” in *2016 IEEE 27th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC)*, pp. 1–6, 2016.

[28] V. Jacobson, D. K. Smetters, J. D. Thornton, M. F. Plass, N. H. Briggs, and R. L. Braynard, “Networking named content,” in *Proceedings of the 5th International Conference on Emerging Networking Experiments and Technologies*, CoNEXT ’09, (New York, NY, USA), p. 1–12, Association for Computing Machinery, 2009.

[29] V. Dimitrov and V. Koptchev, “PSIRP project – publish-subscribe internet routing paradigm: New ideas for future internet,” in *Proceedings of the 11th International Conference on Computer Systems and Technologies and Workshop for PhD Students in Computing on International Conference on Computer Systems and Technologies*, CompSysTech ’10, (New York, NY, USA), p. 167–171, Association for Computing Machinery, 2010.

[30] C. Dannewitz, D. Kutscher, B. Ohlman, S. Farrell, B. Ahlgren, and H. Karl, “Network of information (netinf) – an information-centric networking architecture,” *Computer Communications*, vol. 36, no. 7, p. 721–735, 2013.

[31] A. Ioannou and S. Weber, “A survey of caching policies and forwarding mechanisms in information-centric networking,” *IEEE Communications Surveys Tutorials*, vol. 18, no. 4, pp. 2847–2886, 2016.

[32] S. Zeadally, R. Hunt, Y.-S. Chen, A. Irwin, and A. Hassan, “Vehicular ad hoc networks (VANETS): status, results, and challenges,” *Telecommunication Systems*, vol. 50, p. 217–241, Sep 2010.

[33] M. Amadeo, C. Campolo, and A. Molinaro, “Priority-based content delivery in the internet of vehicles through named data networking,” *Journal of Sensor and Actuator Networks*, vol. 5, no. 4, p. 17, 2016.

[34] C. Yi, A. Afanasyev, L. Wang, B. Zhang, and L. Zhang, “Adaptive forwarding in named data networking,” *SIGCOMM Comput. Commun. Rev.*, vol. 42, p. 62–67, June 2012.

[35] L. Wang, R. Wakikawa, R. Kuntz, R. Vuyyuru, and L. Zhang, “Data naming in vehicle-to-vehicle communications,” in *2012 Proceedings IEEE INFOCOM Workshops*, pp. 328–333, 2012.

[36] D. Pesavento, G. Grassi, C. E. Palazzi, and G. Pau, “A naming scheme to represent geographic areas in NDN,” in *2013 IFIP Wireless Days (WD)*, pp. 1–3, 2013.

[37] W. Drira and F. Filali, “Ndn-q: An NDN query mechanism for efficient v2x data collection,” in *2014 Eleventh Annual IEEE International Conference on Sensing, Communication, and Networking Workshops (SECON Workshops)*, pp. 13–18, 2014.

[38] W. Quan, C. Xu, J. Guan, H. Zhang, and L. A. Grieco, “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*, pp. 1–6, 2014.

[39] Gang Deng, Liwei Wang, Fengchao Li, and Rere Li, “Distributed probabilistic caching strategy in VANETs through named data networking,” in *2016 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, pp. 314–319, 2016.

[40] D. Grewe, M. Wagner, and H. Frey, “Perceive: Proactive caching in icn-based VANETs,” in *2016 IEEE Vehicular Networking Conference (VNC)*, pp. 1–8, 2016.

[41] M. Tarroumi and I. Jabri, “EVNDN: Enhanced vehicular named data networking,” in *2017 International Symposium on Networks, Computers and Communications (ISNCC)*, pp. 1–6, 2017.

[42] 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)*, pp. 1–5, 2016.

[43] H. Maryam, A. Wahid, and M. A. Shah, “Mitigating broadcast storm in interest/data packet forwarding in vehicular content centric networking,” in *2017 International Conference on Communication Technologies (ComTech)*, pp. 162–167, 2017.

[44] S. H. Ahmed, S. H. Bouk, M. A. Yaqub, D. Kim, H. Song, and J. Lloret, “Codie: Controlled data and interest evaluation in vehicular named data networks,” *IEEE Transactions on Vehicular Technology*, vol. 65, no. 6, pp. 3954–3963, 2016.

[45] X. Wang, W. Liu, L. Yang, W. Zhang, and C. Peng, “A new content-centric routing protocol for vehicular ad hoc networks,” in *2016 22nd Asia-Pacific Conference on Communications (APCC)*, pp. 552–558, 2016.

[46] Liang Wang, O. Waltari, and J. Kangasharju, “MobiCCN: Mobility support with greedy routing in content-centric networks,” in *2013 IEEE Global Communications Conference (GLOBECOM)*, pp. 2069–2075, 2013.

[47] Z. Yan, S. Zeadally, and Y. Park, “A novel vehicular information network architecture based on named data networking (NDN),” *IEEE Internet of Things Journal*, vol. 1, no. 6, pp. 525–532, 2014.

[48] 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)*, pp. 15–26, 2017.

[49] Shuai Gao and Hongke Zhang, “Scalable mobility management for content sources in named data networking,” in *2016 13th IEEE Annual Consumer Communications Networking Conference (CCNC)*, pp. 79–84, 2016.

[50] J. Lee, S. Cho, and D. Kim, “Device mobility management in content-centric networking,” *IEEE Communications Magazine*, vol. 50, no. 12, pp. 28–34, 2012.

[51] A. Azgin, R. Ravindran, and G. Wang, “On-demand mobility support with anchor chains in information centric networks,” in *2017 IEEE International Conference on Communications (ICC)*, pp. 1–7, 2017.

[52] J. V. Torres, I. D. Alvarenga, R. Boutaba, and O. C. M. Duarte, “An autonomous and efficient controller-based routing scheme for networking named-data mobility,” *Computer Communications*, vol. 103, p. 94–103, 2017.

[53] M. Hussaini, M. A. Naeem, B. Kim, and I. S. Maijama’a, “Efficient producer mobility management model in information-centric networking,” *IEEE Access*, vol. 7, pp. 42032–42051, 2019.

[54] G. Tyson, N. Sastry, I. Rimac, R. Cuevas, and A. Mauthe, “A survey of mobility in information-centric networks: Challenges and research directions,” in *Proceedings of the 1st ACM Workshop on Emerging Name-Oriented Mobile Networking Design - Architecture, Algorithms, and Applications*, NoM ’12, (New York, NY, USA), p. 1–6, Association for Computing Machinery, 2012.

[55] O. Serhane, K. Yahyaoui, B. Nour, and H. Moungla, “A label-based producer mobility support in 5g-enabled icn networks,” in *2020 International Wireless Communications and Mobile Computing (IWCMC)*, pp. 2094–2099, 2020.

[56] M. Hussaini, S. A. Nor, H. Bello-Salau, H. J. Hadi, A. A. Gumel, and K. A. Jahun, “Mobility support challenges for the integration of 5g and iot in named data networking,” in *2019 2nd International Conference of the IEEE Nigeria Computer Chapter (NigeriaComputConf)*, pp. 1–7, 2019.

[57] Y. Rao, D. Gao, and H. Luo, “Nlba: A novel provider mobility support approach in mobile ndn environment,” in *2014 IEEE 11th Consumer Communications and Networking Conference (CCNC)*, pp. 188–193, 2014.

[58] H. Farahat and H. Hassanein, “Optimal caching for producer mobility support in named data networks,” in *2016 IEEE International Conference on Communications (ICC)*, pp. 1–6, 2016.

[59] F. R. C. Araújo, A. M. D. Sousa, and L. N. Sampaio, “Scan-mob: An opportunistic caching strategy to support producer mobility in named data wireless networking,” *Computer Networks*, vol. 156, p. 62–74, 2019.

[60] M. Amadeo, C. Campolo, A. Molinaro, and G. Ruggeri, “Content-centric wireless networking: A survey,” *Computer Networks*, vol. 72, p. 1–13, 2014.

[61] G. Grassi, D. Pesavento, G. Pau, L. Zhang, and S. Fdida, “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)*, pp. 1–10, 2015.

[62] P. A. Lopez, M. Behrisch, L. Bieker-Walz, J. Erdmann, Y.-P. Flötteröd, R. Hilbrich, L. Lücken, J. Rummel, P. Wagner, and E. Wiessner, “Microscopic traffic simulation using sumo,” in *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*, pp. 2575–2582, 2018.

[63] “ns-3.” `http://www.nsnam.org/`.

[64] H. Farahat and H. Hassanein, “On the design and evaluation of producer mobility management schemes in named data networks,” in *Proceedings of the 18th ACM International Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems*, MSWiM ’15, (New York, NY, USA), p. 171–178, Association for Computing Machinery, 2015.

[65] A. Syawaldipa and S. Haryadi, “Packet drop analysis on ndn’s content distribution, case study: Palapa ring topology,” in *2017 3rd International Conference on Wireless and Telematics (ICWT)*, pp. 195–199, 2017.

[66] O. K. Tonguz, N. Wisitpongphan, J. S. Parikh, F. Bai, P. Mudalige, and V. K. Sadekar, “On the broadcast storm problem in ad hoc wireless networks,” in *2006 3rd International Conference on Broadband Communications, Networks and Systems*, pp. 1–11, 2006.

[67] G. Manzo, E. Kalogeiton, A. Di Maio, T. Braun†, M. R. Palattella, I. Turcanu, R. Souas, and G. Rizzo, “Deepndn: Opportunistic data replication and caching in support of vehicular named data,” in *2020 IEEE 21st International Symposium on ”A World of Wireless, Mobile and Multimedia Networks” (WoWMoM)*, pp. 234–243, 2020.