Universidade de Aveiro Departamento de Electrónica, Telecomunicações e Informática
2019

Luís Miguel Moisés Gameiro

Encaminhamento baseado no contexto em ICNs móveis

Context-based forwarding for mobile ICNs

Universidade de Aveiro Departamento de Electrónica, Telecomunicações e Informática
2019

Luís Miguel Moisés Gameiro

Encaminhamento baseado no contexto em ICNs móveis

Dissertação apresentada à Universidade de Aveiro para cumprimento dos requisitos necessários à obtenção do grau de Mestre em Engenharia de Computadores e Telemática, realizada sob a orientação científica do Doutor Nuno Miguel Abreu Luís, Investigador Auxiliar do Instituto de Telecomunicações de Aveiro e co-orientação científica do Doutor Carlos Roberto Senna, Investigador do Instituto de Telecomunicações de Aveiro.

o júri / the jury

presidente / president	**Doutor Amaro Fernandes de Sousa** Professor Auxiliar do Departamento de Eletrónica, Telecomunicações e Informática da Universidade de Aveiro
vogais / examiners committee	**Professor Doutor Alexandre Júlio Teixeira Santos** Professor Associado com Agregação do Departamento de Informática da Escola de Engenharia da Universidade do Minho
	Doutor Carlos Roberto Senna Investigador do Instituto de Telecomunicações de Aveiro (co-orientador)

acknowledgements

Gostaria de aproveitar esta oportunidade para agradecer, em primeiro lugar, aos meus pais e irmão, pelo seu apoio indescritivel ao longo do meu grau académico e por nunca me deixarem perder a vista dos meus objetivos. Gostaria tambem de agradecer ao Doutor Carlos Senna, por toda a sua orientação e esforço imensurável ao longo desta dissertação, bem como a sua companhia nos momentos mais baixos. Foi instrumental. Gostaria também, de mostrar o meu agradecimento ao Doutor Miguel Luis pela sua orientação e inspiração ao longo deste percurso, e pelos comentários imprescindiveis que me permitiram ultrapassar inumeros desafios. Queria ainda agradecer à professora Susana Sargento pela sua orientação e por me cativar na área de Telecomunicações durante o percurso académico, bem como permitir a minha integração no grupo do NAP. Como não podia deixar de ser, gostaria tambem de agradecer a todos os meus amigos e colegas que me acompanharam ao longo dos ultimos anos, bem como aqueles que conheci no Instituto de Telecomunicações. Vocês foram a família que eu escolhi e nada disto teria sido possível sem a vossa amizade incondicional, bem como o apoio tanto nos momentos de brincadeira como nos sérios. Por último queria agradecer ao Instituto de Telecomunicações e aos projetos MobiWise (POCI-01-0145-FEDER-016426) e InfoCent-IoT (POCI-01-0145-FEDER-030433) financiados através do Fundo Europeu de Desenvolvimento Regional (FEDER), através do Programa Operacional Competitividade e Internacionalização (COMPETE 2020) do Portugal 2020, e Fundação para a Ciência e Tecnologia (FCT).

Resumo

Nas últimas décadas, as redes veiculares ad hoc (VANETs) estiveram na vanguarda da pesquisa, mas continuam a ser afetadas por alta fragmentação na rede, devido à mobilidade contínua dos nós e a sua dispersão geográfica. Para abordar estes problemas, um novo paradigma foi proposto - Redes Centradas na Informação (ICN), cujo foco é a entrega de Conteúdo com base em nomes, sendo ideal para atender ambientes de alta latência. No entanto, as principais soluções propostas para entrega de conteúdo em ICNs não têm em conta o tipo de conteúdo nem as várias interfaces de comunicação disponíveis em cada ponto da rede, factor que pode ser determinante em redes móveis.

O objetivo desta dissertação reside no uso dos conceitos de ICNs para a entrega de informações urgentes e não urgentes em ambientes móveis urbanos. Para isso, foi proposta uma estratégia de encaminhamento baseada em contexto, com um objetivo muito claro: tirar proveito do nome e dados dos pacotes, e da análise de vizinhança dos nós, com vista em fornecer com êxito o conteúdo para a rede no menor período de tempo e sem piorar o congestionamento da rede.

O desenho, implementação e validação da estratégia proposta foram realizados usando o simulador ndnSIM, juntamente com *traces* reais de mobilidade da infraestrutura de comunicação da cidade do Porto.

Os resultados mostram que a estratégia de encaminhamento baseada em contexto proposta para o ICN móvel apresenta uma clara melhoria no desempenho em termos de entrega, mantendo a carga da rede constante. Além disso, através da escolha de melhores caminhos e através da cooperação com mecanismos de armazenamento em cache, é possível alcançar atrasos de transmissão mais baixos.

Abstract

Over the last couple of decades, vehicular ad hoc networks (VANETs) have been at the forefront of research, yet still are afflicted by high network fragmentation, due to their continuous node mobility and geographical dispersion. To address these concerns, a new paradigm was proposed - Information-Centric Networks(ICN), whose focus is the delivery of Content based on names, being ideal to attend to high latency environments. However, the main proposed solutions for content delivery in ICNs do not take into account the type of content nor the various available communication interfaces in each point of the network, a factor which can be deciding in mobile networks.

The scope of this dissertation lies on the use of ICNs concepts for the delivery of both urgent and non-urgent information in urban mobile environments. In order to do so, a context-based forwarding strategy was proposed, with a very clear goal: to take advantage of both packet names and Data, and node's neighbourhood analysis in order to successfully deliver content into the network in the shortest period of time, and without worsening network congestion.

The design, implementation and validation of the proposed strategy was performed using the ndnSIM platform simulator along with real mobility traces from communication infrastructure of the Porto city.

The results show that the proposed context-based forwarding strategy for mobile ICN presents a clear improvement in performance in terms of delivery, while maintaining network overhead at a constant. Furthermore, by means of better pathing and through cooperation with caching mechanisms, lower transmission delays can be attained.

Contents

List of Figures

List of Tables

Acronyms

CCN	Content Centric Network
CS	Content Store
FIB	Forwarding Information Base
FIFO	First-In First-Out
ICN	Information Centric Network
ID	IDentification
IoT	Internet of Things
IP	Internet Protocol
IT	Instituto de Telecomunicações
LCE	Leave Copy Everywhere
LRU	Least Recently Used
NDN	Named Data Network
NLSR	Named Link State Routing
NDNLPv2	Named Data Network Link Protocol Version 2
ndnSIM	Named Data Network Simulator
NACK	Not ACKnowledged
OBU	On-Board Unit
PIT	Pending Interest Table
PFIFO	Priority First-In First-Out
RSU	Road Side Unit
TCP	Transmission Control Protocol
T-R	Time-Restricted
UDP	User Datagram Protocol

URL	Universal Resource Locator
VANET	Vehicular Ah-Hoc NETwork
VCCN	Vehicular Content-Centric Network

Chapter 1

Introduction

The last couple of decades have been the stage for the introduction of many innovative works on telecommunication networks. Terms such as Smart Cities and Internet of Things are constantly on the rise and it is expected that in the near future all types of devices, such as mobile phones or sensors, and many vehicles, such as cars, buses and trucks have the ability to seamlessly communicate with each other, thus forming a vehicular network.

Vehicular ad hoc networks (VANETs) exhibit particular characteristics, such as continuous node mobility and broad geographical dispersion, distancing them from other networks. Therefore, the main challenges that these networks emcopass are relative to the intermittent connectivity and the fluctuating delays in information delivery due to frequent network fragmentation and node mobility. In a unique separate context than that of vehicular networks, a new architecture was proposed with the goal of addressing the efficient and reliable delivery of Content when under high network fragmentation and high latency which could not be readily dealt with by using the well known TCP/IP host-centric communication model.

Hence, a new concept arose - Information Centric Networks [8]. This innovative network paradigm focuses on the Content itself in order to assure a proper data delivery without the limitations imposed by the need to establish a reliable end-to-end connection between hosts. Furthermore, it allows for the appearance of many different architecture designs which follow the paradigm established by ICN, but take advantage of different characteristics contributing for a versatility in architecture to address specific challenges of vehicular networks. This is the groundwork in which this dissertation is developed.

This work is focused on building a forwarding module for ICN-based networks, in order to improve Content delivery in environments afflicted by high node mobility, such as in vehicular networks. The proposed module improve the reliability of the network, taking into account that Data being transmitted can belong to different types of Content and, thus might have different impact in case of Data loss to the entities requesting it. The proposed forwarding strategies have the main role of selecting interfaces to transmit Content under dissemination, from a list of possible paths, in an effort to reach either a Content provider or a requester, while minimizing the use of the network resources available. It is within this context and with the main goal of studying optimized forwarding strategies for an efficient Content delivery in information-centric networks that this dissertation emerges.

1.1 Objectives and Contributions

The main objective of this dissertation is the study and implementation of a context-based forwarding strategies for the delivery of contents through a ICN-base communication infrastructure that includes VANETs. The proposed forwarding strategies extend and adapt concepts from information-centric networks, by means of packet context analysis and neighbourhood evaluation to distribute Content to mobile nodes with maximum efficiency and reliability. Furthermore, the proposed solution should take advantage of the available communication technologies in the vehicular network in order to address mobility steadily, improving the quality of Content delivery across different characteristics. As such, the **objectives** are as follows:

- *survey of related work*: study of literature on forwarding strategies for ICNs to understand the state of the art in this research field;
- *study the current network development software*: evaluate the platforms for developing ICNs in order to choose the best option to use according to the scope of this dissertation;
- *design and implementation of forwarding strategies*: the main objective of this dissertation, is focused on increasing Content delivery ratio, addressing mobility concerns of VANET in a ICN-based communication infrastructure;
- *evaluate the proposed forwarding strategies*: assess the efficiency of the proposed strategies through the use of a real mobile dataset from the Porto city VANET, presenting and substantiating a clear concept and viability of fowarding strategies for ICN using context-awareness;

The answer to the previous objectives and challenges will result in the following **contributions**;

- *development of an ICN-based testbed*: based on ndnSIM [6] improvements were made to the NDN packet structure to encompass context and mobility parameters; changes to the workflow of the NDN architecture to further envelop these parameters; adding a module to consume Data from a real dataset; reformulation of the forwarding module to work with new strategies; redesign of the logging module to adapt it to the new requirements;
- *development of a new forwarding module*: to host the new strategies which takes into account inputs from the network structures as well as context and neighbourhood environments;

1.2 Document Structure

The remaining document is divided in five chapters:

- **Chapter 2** - *Basic Concepts & Related work* - This chapter presents a general analysis and description of the fundamental concepts related to this dissertation: information-centric networks, mobility and Content forwarding strategies;

- **Chapter 3** - *Context-aware Forwarding* - This chapter presents the proposed forwarding approaches to improve Content delivery in mobile ICN, as well as the changes required to the existing architecture;
- **Chapter 4** - *Implementation and Integration* - This chapter presents the implementation and integration of a context-aware forwarding strategy for Information-Centric networks, more specifically, to be integrated in the Named Data Networking architecture;
- **Chapter 5** - *Tests and Results* - This chapter describes the scenarios evaluated, the equipment and the platforms used for the tests. It presents and discusses the results obtained through simulation using real mobility traces;
- **Chapter 6** - *Conclusions and Future Work* - This chapter enumerates the conclusions and points out possible improvements and future orientations that could be developed to improve and complement this work.

Chapter 2

Basic Concepts & Related Work

The main goal of this work is the design, implementation and evaluation of an efficient forwarding strategies following an Information-Centric Network (ICN) approach with mobility awareness to deliver Content between consumers and publishers, using a communication infrastructure with a VANET as testbed. Thus, this work covers several topics from different fields of research. This chapter aims to contextualise the reader with ICN basic concepts and architecture, introduces the concept of mobility from VANETs and its impact in ICN-based networks, and presents the stat of the art about forwarding strategies using concepts of Named Data networks, an extension of the ICN paradigm. This chapter is divided as follows:

- *Section 2.1*: *Information Centric Networking* - describes the ICN architecture characteristics as well as detailing the Named Data Network (NDN) architectural design developed under the concepts of ICN;
- *Section 2.2*: *Mobility* - introduces concept of mobility in communication networks, along with the concerns they introduce both in VANETs and ICNs;
- *Section 2.3* : *Forwarding in NDN* - describes forwarding strategies and mechanisms related to Named Data Networking;
- *Section 2.4* : *Chapter Considerations* - summarizes the chapter.

2.1 Information Centric Networking

The current Internet communication model is built according to a host-centric view of the network. Under this traditional TCP/IP model, the process of requesting a specific Content starts by fully locating and identifying a node in which this Content is stored, followed by contacting the same node for information retrieval, to which a connection is established; this cycle is repeated for every single Content that is requested by a consumer. Such model was designed to connect a limited number of computers, without considering concerns such as native mobility support, in-network caching, and Content-based security, to name a few.

To fulfil these needs, new approaches have been proposed across many research communities [9]. Among them, the Information-Centric Networking (ICN) [10] paradigm has the potential to deal with the key requirements and address the challenges coming from Internet of Things (IoT) and mobility-sensitive environments. ICNs provide an alternative to the traditional end-to-end communication model of the current Internet by placing information

dissemination at the centre of the network-layer design [10]. Conceptually, in ICN, each piece of Data has a unique, persistent and location-independent name that is directly used by the applications for Content search and retrieval, thus enabling the deployment of in-network caching and Content replication to improve the delivery of information. Furthermore, ICN provides Data retrieval based on the request-reply exchange model and native support of multicast, which can be especially useful both in a VANET environment and the IoT.

In ICN the importance lies in the Content as opposed to its location, being that the exchange of the concept of host-based IP addresses with a system of information based on "names" (these being opaque to the network and defined at the application domain) allows network network nodes to search for specific Content stashed somewhere in the network and request that single information, eliminating the need for the node to worry about where the Content is physically located. In addition, they present a dynamic behaviour where, in the case nodes change their location before receiving the desired Data, they are able request the information which could not be acquired in the previous point of access, without the need to try to reestablish "safe" end-to-end connections as in the case of TCP/IP networks [1]. Due to this, they are referred to as "receiver–oriented" networks, where a node makes a request in an upstream direction, and the node that has the desired information replies in the opposite direction with that information (downstream), being usually considered to be *stateless*.

A comparison between the IP and NDN, an architecture which implements the ICN concepts, protocol stacks is depicted in Figure 2.1. Most layers of the stack exhibit bilateral agreements, in which an agreement between two entities occurs; e.g., in the case of NDN, a layer 4 transport protocol is an agreement between some producer and consumer. The sole layer that requires universal agreement is layer 3, the network layer, to which IP attributes much of its success due to its simplicity and the weak demands it makes on layer 2, namely: stateless, unordered, best-effort delivery. NDN's network layer (Section 3) is similar to its counterpart but offers fewer demands on layer 2, giving it many of the same attractive properties [1]. Additionally, NDN can be layered over anything, including IP itself. NDN presents two different layers, strategy and security, which allow it not only to take advantage of multiple simultaneous connectivities, such as Ethernet, Bluetooth and 802.11, to name a few, due to its simpler relationship with layer 2, but also to adapt to changing conditions by exploiting dynamic optimization choices of multiple connectivities.

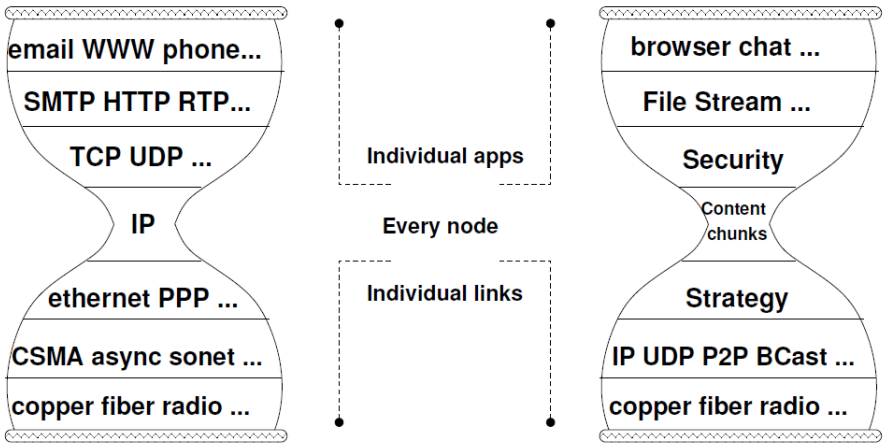


Figure 2.1: IP (left) x NDN (right) Hourglass architecture [1].

In an ICN architecture, the consumer requests Content without knowing the providing host, and the communication follows a receiver-driven approach, *i.e.* Data can only be collected by a reciprocal request from a receiver entity. The system is then responsible for mapping the requested Data and its location. Each ICN architecture is characterised by the adopted approach on each of the following ICN key functionalities: naming and name resolution, caching, and routing and forwarding [11].

2.1.1 ICN Naming

As one of the cornerstones of the ICN paradigm, **naming** refers to the assumption of information being titled, forwarded and matched separately from its physical location, under the prospect of being stationed anywhere in the network, eliminating the need of establishing a source-destination host pair communication [4, 12]. Existing ICN naming schemes can be categorized into four different types: hierarchical, flat, attribute-based and hybrid.

- **Hierarchical**: as the name implies, this scheme assumes the identification of Content though hierarchically structured names, usually in a human-readable format, similar to URLs or current file systems. This allows aggregation across similar names, which boosts the scalability potential of the architecture by reducing routing tables size. The most common architecture using this scheme is NDN [3], in which a name is arbitrary in form and length;

- **Flat**: flat names usually consist of fixed, short and globally unique strings, which are self-certified. These are commonly computed using hashes of either the full Content or some part of it, normally being non-human readable in nature [13]. The adaptation of a fixed-length name results in the consumption of less storage space in the cache, as well as a more efficient lookup through structures such as Bloom filters, however will difficult aggregation which makes routing tables more complex.

- **Attribute-based**: aims to capitalise the nature of certain Content by extracting attributes which are directly related to it under the form of keywords. A Content attribute can be comprised of information such as type, versioning, location or any other specific traits which is used by by routing mechanisms to associate Content with a specific request [14]. This scheme thus promotes an easier search for Content, by moving from full-on name lookup to inquiry of keywords that are associated with the Content;

- **Hybrid**: is characterised by combining the previous schemes to take advantage of the most notable parts of each other [15]. One such case, [2] proposes a hierarchical and hash-based naming with efficient Compact Trie (CT) management scheme for Vehicular Content-Centric Network (VCCN), as illustrated in Figure 2.2. The hierarchical part of the digital Content name contains information about the Content provider and attributes of the Content itself that is shared between vehicles. The hash part uniquely identifies the digital Content required for VCCN applications.

All these categories can be aggregated to some level, improving routing scalability. However, replication and mobility of Content in the network can affect the aggregation and increase the routing table size even with a hierarchical name schema. For most current solutions, hierarchical names are handled semantically at the network layer, where they are also managed,

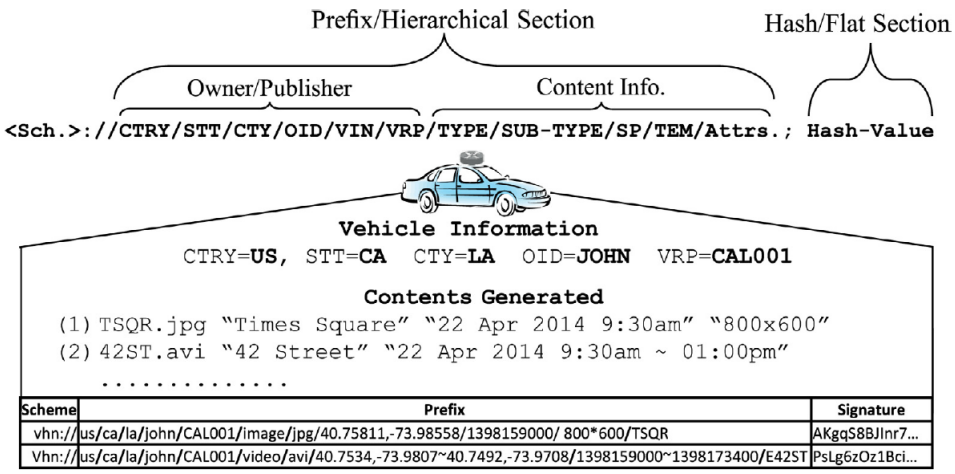


Figure 2.2: Hybrid name (hierarchical & flat) [2].

while schemes that use flat names are more closely linked to the application layer, as they can use their characteristics to generate unique names more efficiently. Regardless of the scheme adopted, naming affects performance and scalability concerning forwarding and tables size. For IoT environments with constrained devices, these implications may be more severe [16].

2.1.2 ICN Caching

Caching refers to storing Content temporarily to serve it on future requests closer to consumers, reducing the number of messages that need to be transmitted between a consumer who requests a particular Content and the producer which provides it [17]. It can also reduce a response bottleneck by having a Content distributed by the network, increasing availability and decreasing latency. However, this procedure turns out to be complex as the decisions taken on where to place Contents efficiently and how to manage them when the cache needs to evict some Content over another have a direct impact on how the network performance. It is also important to determine if it is more advantageous to perform storage in caches along the path (*on-path caching*) or any other caches (*off-path caching*).

This process is controlled by a caching strategy which can have different characteristics according to three different phases: placement, replacement and coherency.

- **Placement** is about deciding where and when Content should be stored. This choice is based on Content placement schemes like Leave Copy Everywhere (LCE) [3], or Probabilistic Caching (ProbCache) [18], among others. The LCE strategy indicates that all nodes through which Content goes through will store it without any restrictions. ProbCache proposes to store Content following an inversely proportional relation between the distance from producer to consumer, thus obtaining a probability value;

- **Replacement** is about determining which cached Content should be replaced by a new Content when the cache reaches its maximum capacity. There are several Content replacement schemes like First In First Out (FIFO) and Least Recently Used (LRU), among others. In FIFO, the Content that came in first of all is chosen as the cache victim to be replaced. LRU selects the Content that has not been used the longest, eliminating Content on low demand.

- **Coherency** is about checking the validity of the cached Contents. Coherence mechanisms are focused on verifying if the Content is still up to date.

Cache placement can have a critical impact on the performance of a network due to its potential to increase Content availability, however, its proper distribution must be ensured so as not to cause redundancy. Together with replacement policies, they may increase or degrade the efficiency of a caching strategy, so it is necessary to reconcile the best decisions of both phases, as well as ensuring coherency of the cached Content.

2.1.3 ICN Routing and Forwarding

As referred in the beginning of the Section 2.1, by using names as the focal point of forwarding emphasis, it is possible to support seamless mobility without encompassing all the complex network management required by IP networks. In a wide view of forwarding process, the pending Interests state at each router supports Data forwarding across ICN, recording an association between each pending Interest and the incoming interface(s), removing the Interest after the matching Data is received or a timeout occurs. Based on forwarding information and performance measurements, each router decides which Interests to forward to which interfaces, how many unsatisfied Interests, the relative priority of different Interests, load-balancing Interest forwarding among multiple interfaces, and choosing alternative paths to avoid detected failure [3, 19, 20].

If a router decides that the Interest cannot be satisfied, e.g., the upstream link is down, there is no adequate forwarding information, or extreme congestion occurs, the router can send a NACK to its downstream neighbour(s) that transmitted the Interest [20]. When receiving a NACK the receiving router forwards the Interest to other interfaces to explore alternate paths. The analysis of the pending Interests registry enables routers to identify and discard looping packets, allowing them to freely use multiple paths toward the same Data producer. By making the proper decision as to how a Content should be transmitted, Content availability is increased while reducing delays resulting from the ensuing communications which boosts connectivity overall.

The most notorious properties of ICN routing under analysis, as described in [4], are:

- **Content state**: routing mechanisms should administer low latency operations, such as Content registration, lookup and deletion, to accurately maintain Content state properties, such as freshness and availability, within the network;

- **Discovery of closest copy**: effort should be employed by the routing process to ensure that a request for Content is resolved by the discovery of its nearest copy according to an established network metric, thus reducing traffic in the network;

- **Discovery guarantee**: existing Content should be given equal relevance, in spite of its popularity or replication, therefore guaranteeing delivery of Data requested by consumers;

- **Scalability**: routing mechanisms need to be able to support such magnitude of names which trend is to grow over time. Routing table size and routing path length should be kept low to prevent possible bottlenecks within the network.

2.1.4 ICN architectures

Over the past years, several ICN architectures have been proposed[10]. However, the vast majority solely address infrastructure networks, in an effort to better cope with issues such as disruptions and flash crowd effects, without considering mobility nor the requirements of an IoT environment. Moreover, due to the versatility of ICN paradigms, their designs present unique capabilities which are usually focused on tackling specific concerns as opposed to a more general approach. The most significant architectures are briefly described below. For the Named Data Network solution, the base architecture used in this dissertation, a detailed description is presented.

2.1.4.1 Data Oriented Network Architecture

Data Oriented Network Architecture (**DONA**) [21] proposes the use of a flat and self-certified naming scheme under a resolution infrastructure. In this design, Content providers are required to be authorised to serve Data as well as being registered to a resolution-handler which acts as a gateway between the Data source and a Data requester. These handlers route the Content hierarchically according to its name, and have the capability of storing it in cache in order to ensure future requests for the same Content are resolved closer to the consumers. Flat names are under the form P : L, where P is the unique field which contains the cryptographic hash of the publishers public key and L is the unique object label. Due to this format, a Content being published by a different entity than where it was originated from, such as a Resolution handler, possesses a different name even though it refers to the same Data, which can make taking advantage of available copies more difficult and is not plausible in terms of scalability.

2.1.4.2 Network Information

Network Information (**NetInf**) [22] offers two models for retrieving Data in a hybrid format: via name resolution or via named based routing. On one hand, Content providers or nodes holding a copy of the Content can register names with a name resolution service, which a Content requester can make use of by resolving a name into a set of available locators and therefore retrieve the desired Content from the optimal source. On the other hand, the consumer directly send a request for a certain Content by means of a name-based routing protocol which will forward the request towards an available copy. The use of these two models makes NetInf very versatile, by ensuring its use in different network environments, in which one model is more suitable than the other. NetInf makes use of flat, self-certified names, which do not provide scalability.

2.1.4.3 Publish Subscribe Internet Technologies

Publish Subscribe Internet Technologies (**PURSUIT**) [23] implements the ICN concept using a publish-subscribe network model. The naming scheme adopted here is the same one proposed by DONA which follows the P : L flat format. PURSUIT makes uses of special dedicated entities referred to as "Data sources" which periodically refresh Content publication states and maintain information regarding the Content providers (*publish*) and consumers requesting the Content (*subscribe*), by means of a pair of IDs connecting the two *(Source ID, Requester ID)*. In order to make it possible for consumers to subscribe to advertised Contents,

PURSUIT makes use of a Rendezvous system, in which a subscription request is matched with the Data source according to a unique name. The connection that ensues when a new Content is available is then performed by the Data sources by forwarding the new Data to the consumer which is subscribed to it.

2.1.4.4 Mobility First

Mobility First (**MF**) [24] assigns a flat, globally unique identifier to every network object, separating the identifier of the object from its network address, which allows for a good mobility support by means of dynamic address binding. MF makes use of three key components to support its core architecture: a Global Name Resolution System charged with managing all the dynamic mapping between network address and object identifiers, as well as making sure these values are updated when a mobile node changes its network association; a hybrid routing protocol in which routers routers perform decisions based on both unique identifiers and network addresses, in a hop-by-hop basis; and delay-tolerant networking by means of caching Data packets in the routers.

2.1.4.5 Named Data Network

Many of the recent studies have been focused on what is known as Named Data Network (NDN) [3], which stands for the use of hierarchically structured Content names, Content source reliability (by means of authentication by the Content provider through cryptographic mechanisms such as public keys) and a routing protocol which performs routing mechanisms using names in place of IP addresses. This section presents a small overview of the NDN architecture, names and packets, basic structures and forwarding strategy. For a deeper look at this project see [1, 3].

NDN ***names*** are opaque to the network. The NDN design assumes hierarchically structured names. For example, cs.uwaterloo.ca/r5ahmed/drft_naming.pdf/_v1/_s1 (Figure 2.4 (a)) similar to URLs. This hierarchical structure allows applications to represent the context and relationships of Data elements.

The NDN architecture uses two types of packets in all communications: the **Interest** packet, used to define a request for Content, and the **Data** packet, containing the Data itself. Both have a common Content identifier designated as **Name**, which is a sequence of name components that follow a hierarchical structure with variable length.

The **Interest packet**, shown in the left side of Figure 2.3, contains two required parameters known as *Name*, which identifies the desired Content, and *Nonce*, which identifies uniquely the packet travelling through the network. It has also several optional parameters, aggregated as *Selectors*, which serve as auxiliary input to the forwarding or Content resolution mechanisms, such as indicating if the Interest can be satisfied with stale Data; accessing the lifetime duration until the Interest expires, whose timeout is related to the arrival time of the packet at the current node; the hop count limit before an Interest is dropped, among others.

The **Data packet**, shown in the right side of Figure 2.3, contains the following parameters: a *Name*, which identifies the desired Content, a *Signature* and *SignedInfo* fields, which contain security information necessary to validate the Content and which relate to the publisher; and the *Data* itself. The creation of a Data packet is directly associated with the reception of an Interest counterpart, so the "Name" field is always extracted from the received request and thus is the same.

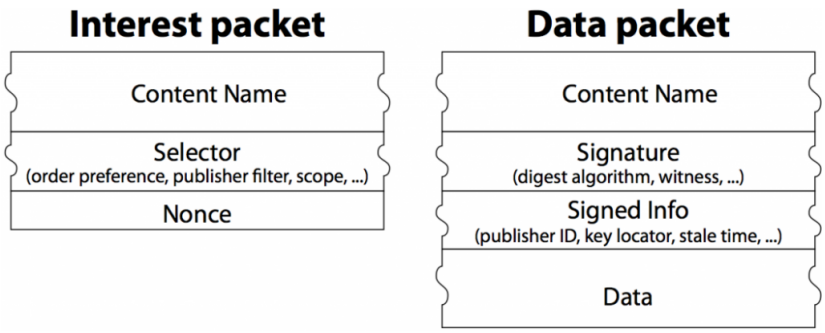


Figure 2.3: Packet structure in NDN architecture [3].

The devices exchanging information in the NDN architecture are treated as nodes that integrate the NDN logic, and can behave either as ***producers*** (entities originating and signing Contents), ***consumers*** (entities who issue Interests for certain Contents) or ***router/intermediate nodes*** (entities which forward Interest packets and route Data packets between consumers and producers). A node can assume the three roles referenced concurrently.

A NDN node maintains three basic structures known as *Content Store* (CS), *Pending Interest Table* (PIT) and *Forwarding Information Base* (FIB), as illustrated in Figure 2.4(b). The CS is a cache for incoming Data packets, which may be used for faster replies to Content requests. The PIT is a registry table of Interests and respective interfaces (*faces*) from which they came and which are waiting for the requested Data. The FIB is a table of name prefixes with respective outgoing *faces*, which allows the routing of the Interest packets through a name-based lookup. The component known as ***faces*** enables to receive and forward packets, thus it is seen as an interface with appropriate mechanisms.

Packet forwarding in NDN is achieved by exploration of up/down streams in each NDN router according to the information available in the Data structures (CS, PIT, and FIB). Each Interest packet contains the name of the Content to achieve alongside with any useful network Data relevant information to the future intervening nodes. It is sent to the network, being forwarded along established paths that are present in its FIB. The basic algorithm is illustrated by diagram in Figure 2.4 (c). When an Interest reaches a node, it checks its CS for the Data packet. If it is not in the CS, it proceeds to the PIT where it stores the Interest along with a registry of the interface from which it arrived, and then makes use of the FIB to route the Interest packet, if possible. When the Interest reaches a node in possession of the requested Data, that node replies with a Data packet containing the corresponding Content, along with a signature done by the producer. This Data packet follows the path taken by the Interest packet in reverse, returning to the consumer which made the request. This also applies to the NACK packets that notify errors like congestion, Interest duplication or no route found for that Data packet, which also follows the taken path to the consumer. This processing of the forwarding of Interest and Data packets is illustrated in Figure 2.5.

NDN uses a link protocol known as NDNLPv2 [1]. It is designed to be used with all types of links such as Ethernet unicast/multicast, UDP unicast/multicast, TCP connections, among others. It operates as a link adaptation layer, located between link and network layers. It provides several features, which are all optional and can be modified per-link: packets fragmentation and reassembly, failure detection, reliability, integrity, forwarding instruction

[1] https://redmine.named-data.net/projects/nfd/wiki/NDNLPv2

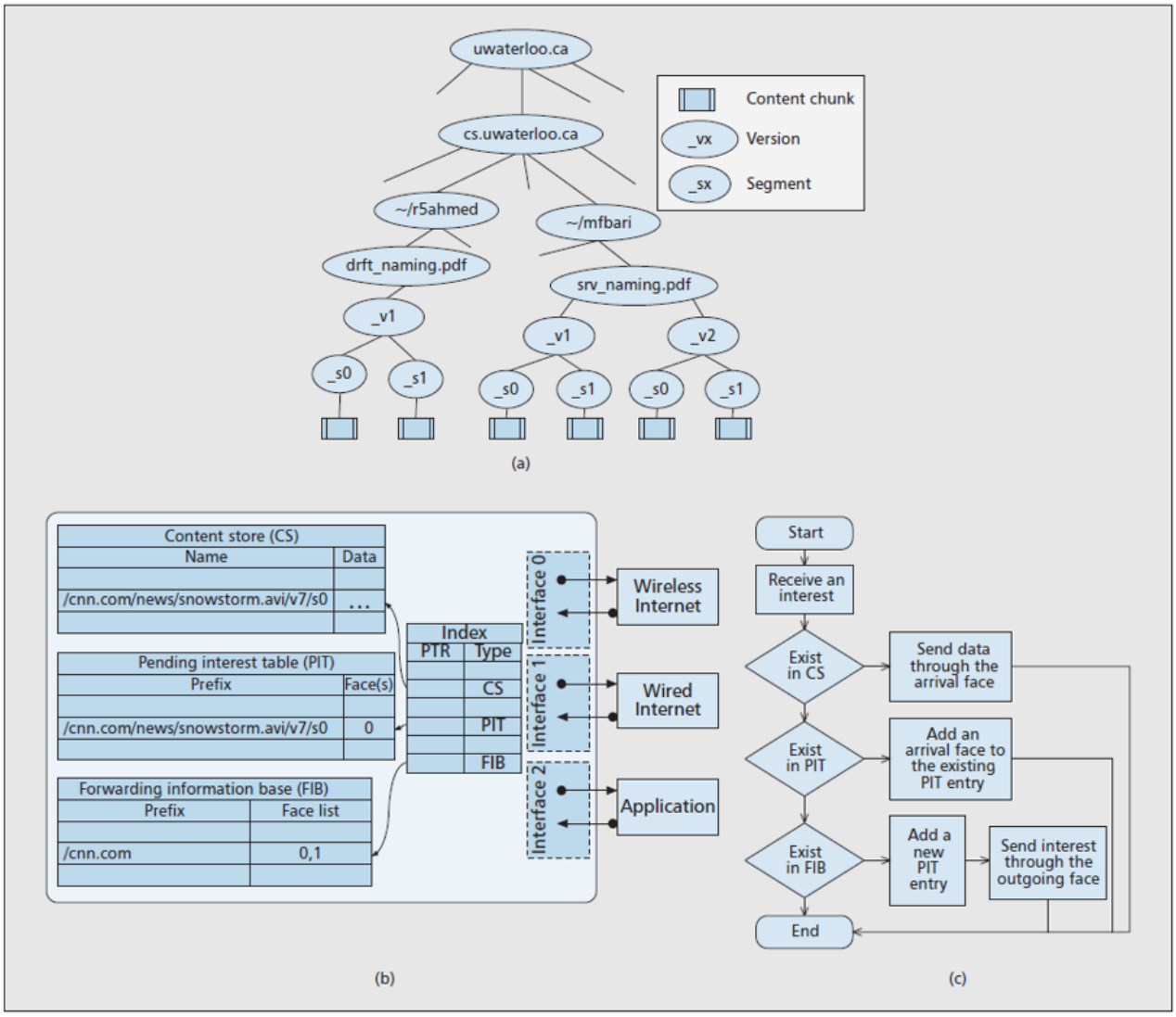


Figure 2.4: Processing of Data in NDN. a) Architecture; b) Internal structures; c) Processing of Interests [4].

and packet information. The NDNLPv2 packet, known as *LpPacket*, carries NDN packets and can also contain additional information which is represented by tags encoded into the header field. Being a single-hop protocol and with the forwarding plane not obliged to preserve tags, it is beneficial to use tags in cases where it is necessary to update these parameters along a path. The communication between network nodes occurs using Interest and Data packets, which are encapsulated using the link protocol NDNLPv2.

2.2 Mobility

In a wired infrastructure, devices are bound by the constraints imposed by the need to maintain a connection between the entities. In an ad-hoc network, however, devices make use of ranged technologies to communicate with each other, thus being free to move. In order to

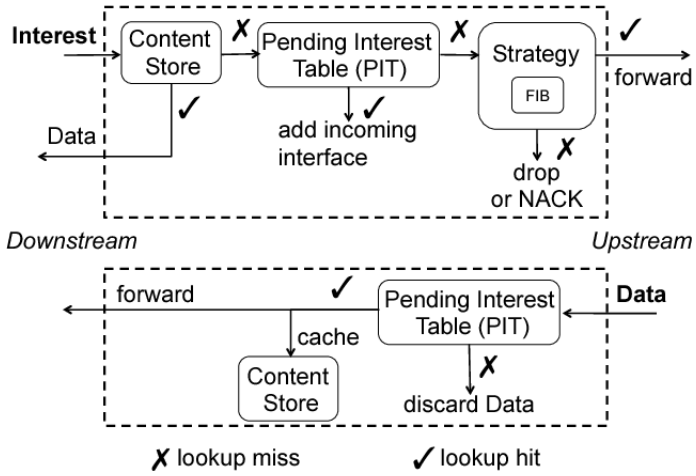


Figure 2.5: Forwarding process in NDN.

support mobility, a stable connection needs to be kept alive throughout the user's movement in the physical environment. In a traditional TCP/IP network, a handoff occurs when a mobile host moves from the neighbourhood of a base station or access point to another one. A protocol is therefore required to ensure a seamless transition during said handoff, which includes deciding when it should occur and how Data is redirected during the process. In the case of an ad-hoc network, the topology is constantly changing due to the intermittent connectivity of mobile nodes entering and leaving the network. Due to the fact this type of network may be comprised of a large number of mobile hosts, this poses great emphasis on the design of an efficient routing protocol that can work well in an environment with dynamic characteristics caused by frequent topological changes.

The established IP address-centric model exhibits many difficulties in such extremely dynamic situations, which is specially evident in vehicular communications, which usually target specific road areas and an unpredictable amount of vehicles, regardless of the identity of each moving node. Thus, a shift towards the use of ICN-based paradigms has been the core focus of research community to address this concern, which is further raised due to the continuous evolution of the concept of Smart Cities and their aim to provide seamless connectivity between hosts and various applications in their neighbourhood. In this degree, its vital to understand the characteristics of a network comprised of vehicles and how the ICN design can be applied to complement it.

2.2.1 Vehicular ad-hoc networks

A Vehicular Ad-hoc Network (VANET) [25, 26] is a network that is composed of vehicles equipped with wireless interfaces which allow communication between them and fixed infrastructures installed along the road in strategic locations (as illustrated in Figure 2.6). The nodes of a VANET are equipped with an On-Board Unit (OBU), associated with mobile nodes (vehicles), which is a device equipped with a wireless communication interface to connect to other OBUs as well as a wired communication interface to allow communication with applications in order to collect Data. Thus, these units compose the mobile part of the communication within the network and allow dissemination of Data between each other.

The OBUs communicate with the infrastructure through a Road-Side Unit (RSU) which is

usually a static node. These devices are likewise equipped with two or more communication interfaces which serve as a bridge between mobile nodes and the network infrastructure. These type of networks can be deployed using a wide variety of vehicles such as private cars, ambulances, buses,taxis, bikes, and drones.

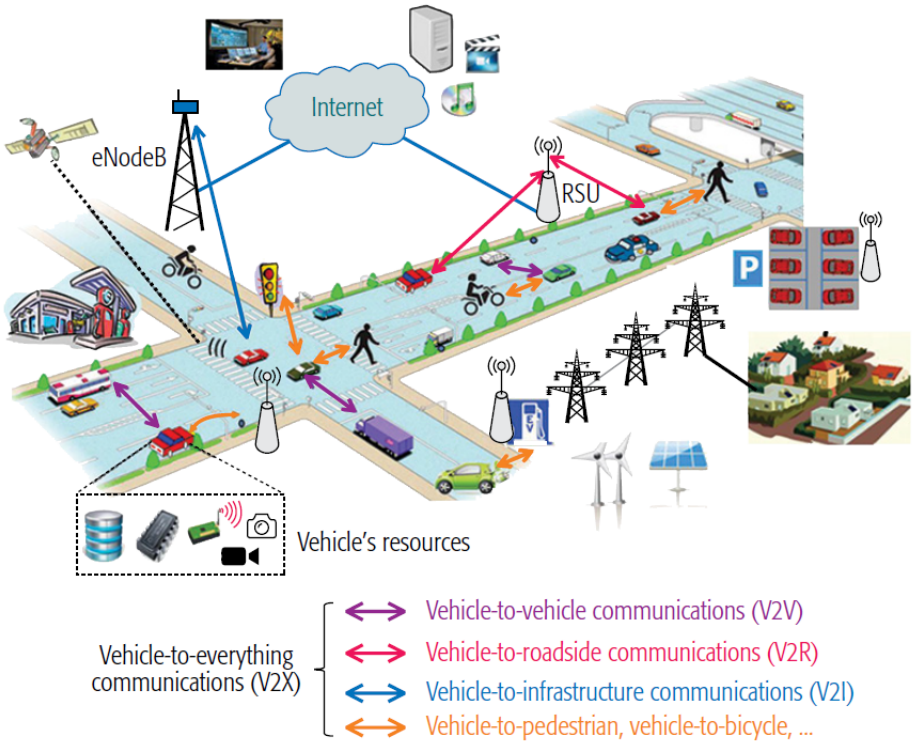


Figure 2.6: VANET and vehicular communications in Smart City context [5].

The colored lines of Figure 2.6 highlight the types of communication commonly used in VANETS [27].

- **Pure Ad-Hoc (V2V)**: There is only vehicle-to-vehicle communication (V2V). A node reaches other nodes using neighbouring vehicles without any infrastructure support.
- **Vehicle-to-Infrastructure (V2R/V2I)**: OBUs communicate with RSUs in order to access other networks and farther nodes.
- **Mobile Infrastructure**: Some OBUs can serve as Mobile Gateways in order to allow access to other networks.

This versatility in the communication model translates to a network environment which can present a different behaviour across a wide variety of scenarios in which different protocols can be applied to address specific constraints of each model.

VANETs are characterized by the main following aspects, according to [28]:

- **High Mobility**: The most important characteristic. Nodes are in constant movement with different speeds and towards different directions. The mobility of the nodes cause the reduced contact between nodes which can result in less available paths between them.

- **Dynamic Topology**: Due to the high mobility, VANET topology suffers constant changes resulting in unpredictability. Nodes that move in opposite directions are burdened by reduced contact times. This promotes increased difficulty in terms of identifying malfunction within the network.
- **Frequent Disconnections**: The fact that nodes are highly mobile and may be under conditions which harden the communication between them, such as climate conditions or the presence of buildings in between the intervening entities results in frequent disconnections in all mobile nodes.

While VANETs may represent a sound solution to tackle mobility concerns, the challenges related to the intermittent connectivity and the resulting long and variable delay in information delivery still need to be addressed when the focus is an efficient and reliable delivery of Content to consumers.

2.2.2 Mobility in ICN

In an ICN context, whenever a consumer expresses interest in a particular Content, it issues a request that is forwarded in the network until it reaches a viable Content provider which replies with the requested Content in-reverse path. Throughout this whole process, publishers and consumers lack the need to have spatial references to each other and are not aware of how many publishers and consumers are involved in a specific Content delivery process. This decoupling of space and time between the Interest delivery and Data transfer itself is one of the basic features of this design that allows for seamless mobility of mobile nodes.

When used in conjuction with the NDN paradigm, broadcasting is generally used to transmit NDN packets [29], mainly due to its inherent advantages: (i) it allows reachability to multiple nodes, potentially storing the Content of Interest, with a single packet transmission, and (ii) it may facilitate Content sharing, since broadcast packets can be overheard over the wireless channel. Broadcast may unavoidably result in high channel access Contention, several collisions and redundancy, especially in dense multi-hop environments. Undoubtedly, a broadcast request may be received and processed by several nodes at once. Some of them could simultaneously satisfy the request and reply with a Data packet, while the other nodes without the Content copy will forward the Interest ahead. These issues need to be addressed if the aim is an efficient Content delivery, due to the network overhead introduced by the broadcast method, which is specially prevalent for delay-sensitive scenarios.

ICN design inherently backs consumer mobility support [30] through quick Interest retransmissions whenever a mobile node changes neighbourhoods and is not able to acquire the requested Content as a whole in the previous point of access. This introduces a higher number of Content requests in the network, which promotes congestion while reducing the burden on the routing and forwarding mechanisms to ensure Data is delivered to the new physical location of the mobile node. Moreover, caching policies help in the process by moving Content to locations near to the consumers and therefore reply quickly to future requests. However, this can also infer redundancy of Content in the network due to Content being cached in locations in which current nodes may no longer be interested in.

To this extent some research has been conducted to address consumer mobility in a more efficient manner for ICN environments. In [31], Han *et al.* design a new retransmission method to estimate an adequate lifetime for expired packets and therefore control how often

retransmissions occur. Due to factors such as in-network caching, not only may certain Content lifetime fluctuate over time, but also the delays or losses associated with its retrieval vary, thus the authors explore the importance to control the retransmission window size for a given Content in an effort to diminish packet loss caused by consumer mobility. Amadeo *et al.* [32] explore a Content retransmission recovery method by means of better handling how that procedure is triggered in the first place. In CCN, if a given Data is fragmented in several chunks and loss occurs during the transmission to the consumer, an Interest retransmission is triggered for the whole Content, without proceeding to an analysis of how many chunks were received successfully, causing unnecessary packet retransmissions. Thus, the authors propose that retransmission should be selective, studying which Data chunks were not received and therefore need to be requested, instead of the whole Content. Furthermore, this process is prompted only after a reasonable time window, meaning that the consumer has to wait for all Data chunks in a Data reception to arrive, and will only issue re-transmit Interests for specific chunks after expiry occurs.

In spite of supporting native consumer mobility via Content retransmissions, ICN does not solve the concerns introduced by the delay to received requested Data when a mobile consumer moves to a new location, which is specially prevalent for delay-sensitive applications. To this extent, the concept and application of proactive caching methods for ICN is also given attention by some researchers. In [33] consumer mobility is explored under a *publish-subscribe* environment, by considering the probability of a subscriber's movement pattern and selecting specific caches in the future neighbourhood of the mobile node to store the Content being requested at that instant, resulting in the Data being fully available to that node during its movement without inferring any delays. However, this selection needs to be criterious due to the possible impact of choosing incorrect caches to store the Content and the redundancy of Content in the network.

There are also some unique approaches that make use of how forwarding works to address specific mobility concerns. In [34, 35], Kim *et al.* propose to manipulate the lifetime of Interest packets in CCN, by means of a new *persistent packet* with the aim of addressing time-constrained services, such as video-streaming. These Interests would remain in the PIT structure until a signal is sent by the Content provider informing that the Content is no longer available, resulting in the reduced need for the consumer to send Interests every single time a new Content is available for that specific name, as well as promoting the choice for a path with more throughput when available and if deemed necessary, since the Content is always available until further notice. However, the challenge lies in choosing the most adequate nodes to have persistent Interests stored in the PIT, which impact directly how delays in the Content retrieval occur.

On the other hand, due to the receiver-driven nature of ICN, publisher mobility is tougher to handle than its counterpart since, when a Content provider is on the move, the paths the Interests that were being forwarded across to the previous publisher location become obsolete and the network domain structures need to be updated to reflect those changes in an attempt to deliver the requests. This issue is especially prevalent when the requested Data is time-constricted, thus needing to be resolved reliably within a limited period, and much community research has been focused in this regard to this day. In [34, 35], two possible methods are presented to address this concern. The authors propose a routing-based approach to address this issue by using the NDN's Interest packets to support producer mobility. When a producer is on the move and a neighbourhood change is imminent while a communication is ensuing, a virtual Interest packet is sent to the previous intermediate node in order to update the

FIB structure, which stores the next hops to reach the Content for a given name, so that the paths are updated seamlessly. This information is then forwarded across the intermediate nodes until it reaches the consumer, instant at which the path is correctly updated to reach the Content provider. The authors also propose an indirect approach for ICN, by means of a rendezvous point to provide name resolution between names and associated Content providers, which is updated whenever a producer moves to a new location. This point ,thus, is responsible for forwarding Interest packets to the producers, but require some kind of location information to be stored locally instead of resolving Content on the fly. This last method, however, promoted higher latency due to the name resolution mechanism.

2.3 Forwarding in Mobile NDN

The main focus of this dissertation lies on forwarding strategies for mobile ICN environments. Thus, in this session we depth discuss the concepts and work related to ICN forwarding, already applying.

The transition from an end-to-end communication model to one focused on the Content itself and its delivery based on names, paves the way for many new improvements to how requests are attended to, with emphasis on the forwarding strategies, which are capable of dynamically selecting network interfaces according to a given node network conditions, such as delays or congestion status. Much research has been dedicated to ensure Interests are transmitted efficiently in the network environment with attention on issues such as redundancy, bandwidth usage and delay management. According to [36], forwarding strategies can be divided into four types: Congestion control, adaptive forwarding, blind forwarding and aware forwarding.

- **Congestion control**: Forwarding strategies under this type are designed with the aim of regulating packet transmission flow by means of Interest shaping in order to reduce network load and increase link utilisation. In [37] Wang *et. al* points that both *Interest* and *Data* packets contribute to congestion and proposes a congestion control scheme to regulate *Interest* packet transmission flow, by means of studying Content demand on a per-node basis to weigh in the forwarding decisions. Moreover, the study makes a statement that *Interest* packet drops have less impact in the communication when compared to *Data* packet drops and reveals that having a preference for Interest removal in case of congestion results in higher network throughput. In [38], Jiang *et. al* takes advantage of NDN Interest aggregation capabilities and reveals that grouping *Interest* packets with the same flow contributes to the reduction of network overhead, round-trip time delays, reduced FIB lookups and also reduction in PIT sizes. These proposed solutions have some limitations, however, by assuming that network and traffic properties like the size of Data packets, the time required to retrieve Data packets and the bandwidth of each link, are known beforehand.

- **Adaptive Forwarding**: states that the strategy accommodates different network conditions which may change overtime, striking a balance between several performance metrics and context awareness to ease possible route decisions. In [39] Yi *et. al* state that in NDN routing behaves the same as IP while forwarding reacts adaptively. To this extent, the authors propose a probability-based adaptive forwarding (PAF), which makes use of an ant colony optimisation technique to calculate each interface selection

probability, reducing delays as well as detecting network changes, such as topology alterations or link failures, by means of efficient load balancing. Moreover, this strategy can be turn on or off according to the network conditions, further augmenting its adaptability. Zeng *et. al* [40] propose a rank-based strategy for interface selection according to interface load. Outgoing interfaces which are deemed busier are attributed lower rank than the remaining ones, promoting load balancing across the the different available interfaces, thus reducing the load in the node. Yi *et. al* [41] improves upon this concept by including Data delivery in the ranking process, ensuring that interfaces with a history of delivering the Content successfully are attributed more ranking points.

- **Blind Forwarding**: strategies following this scheme are usually aimed towards Interest flooding and network overhead control by means of Interest packets flow limitation. Wang *et. al* [42] advocates the use of timers to control packet transmission in conjunction with nodes listening to the medium to decide if they should send the Interest or Data packets. In case a node overhears a certain Content which he also owns, ready for transmission, being transmitted to the medium, a decision is made towards remaining silent, due to it already being disseminated, which reduces network overhead. Amadeo *et. al* [43] propose a similar concept by means of a counter-based approach instead of a timer-based one, prioritizing Data delivery over Interest propagation. These approaches, however, do not guarantee selection of the best node to send packets and are not guaranteed to solve packet collision by use of overhearing the medium. Moreover, the use of timers is not suitable to address broadcast storm issues.

- **Aware Forwarding**: strategies under this scheme maintain knowledge of certain external network states, such as neighbour nodes, in order to weigh in on forwarding decisions. In [44], Lu *et. al* propose a Next-hop awareness scheme, in which a node replying with Data transmits in broadcast mode to its 1-hop neighbours, and the farthest neighbouring node in that quadrant is selected to continue forwarding the packets. Yu *et. al* [45] present a neighbourhood awareness scheme, in which nodes forwarding Interests by selecting their neighbours based on their delivery ratio for each prefix in the network. If that value is too low or the neighbour node is far away, the Interest is discarded locally, preventing unnecessary network overhead. In [46] a provider aware strategy is proposed, in which a node requesting Data has knowledge of and selects a provider, including its ID in the Interest packets. This ID is used by the intermediate nodes in order to forward the Interest towards the selected provider according to a best-route mechanism; likewise, the Data requested will travel back to the Content requester in a best-route manner. Moreover, a timer is used in order to control Data redundancy in the network. These approaches promote better forwarding decisions by adapting to different network conditions, and topologies to an extent, but require more resources than their counterparts due to nodes needing to maintain information about their awareness selected metric.

While most of the previously mentioned works improve upon NDN native forwarding mechanism, there are also some which include their own unique changes which prevents them to be directly inserted into any given scheme. In [47], Garcia-Luna-Aceves *et. al* replace the PIT structure with Data Answer Routing Tables (DARTs) which keep route segments of the path taken by the Interest towards the Content provider as opposed to the Interest itself. Thus, its size is proportional to the number of routes taken by the Interests instead of

the number of unique Interests in the network. In order to address the Interest loop issue, DARTs compare the distance between announced prefixes and their neighbours in order to understand if a packet should be dropped, resulting in a more efficient storage overhead.

Zhang *et. al* [48] state that the use of parallel multi-pathing is benevolent to the throughput of the network, but which might result in an increased cache pollution. To this extent, the authors propose a method to achieve that parallelism without causing pollution by means of Luby Transform (LT) coding. In [49], Duarte *et. al* propose a probabilistic Interest forwarding strategy aimed towards delay tolerant networks, by studying the frequency a node comes in contact with a given Content as well as its freshness in order to establish a Delivery Predictability (DP) value for all available nodes towards that Content, thus, the one which presents the highest DP value is selected as the receiver of the Interest due to the higher probability of delivering the Content.

Tsilopoulos *et. al* [50] present an adaptive semi-stateless forwarding scheme where Interests are tracked only on a fraction of the routers as opposed to all or none as in traditional NDN. Between state-tracking routers, Interest packets gather reverse-path information, which is used to deliver Data via Bloom filter-based forwarding, instead of using the incoming interface, as per search in the PIT structure. This whole process reduces the burden on routers resources by splitting the needs of the forwarding method across several nodes.

Different NDN packet forwarding mechanisms have been proposed, but few of them have considered the idea of a prioritised traffic management based on Content type that will be disseminated in VANET. Amadeo *et. al* [29], propose a priority-based Content dissemination scheme for vehicular environments in order to meet the requirements of heterogeneous applications. Based on the NDN hierarchical namespace, they proposed specific "name-prefixes" that identify globally understood priorities for vehicular Data traffic.

As a further matter, most works on routing for ICN architectures are designed using the traditional strategy to forward Content requests for the Content producers through a best-route approach, not aiming at the efficient support for multi-path forwarding [51], which has the potential to boost the performance of Content delivery within the network massively if implemented correctly.

The main focus of this dissertation is to address some of these issues by improving the NDN forwarding workflow for both Interest and Data packets in a manner that takes advantage of each node's available communication technologies and properties, as well as understand the context in which a given prefix is inserted into, to more efficiently resolve Content.

2.4 Final Considerations

In this section the concept of ICNs was presented. According to the given description, the positive impact of this architecture design in today's society is very relevant leading to a new set of possible applications and services in a world of increasing mobile connectivity. This innovative network architecture is based on the notion of forwarding Content by name identifiers which decouple the devices from the need to establish host-centred connections and thus support native mobility. Therefore, networks designed under this spectrum represent an environment with a high potential to provide and support Content delivery through a significant area of Interest.

The concerns related to the introduction of mobility in a network were described, and the concept of a VANET were also presented. This network architecture is based on vehicles

which interact with each other and the fixed infrastructure in order to provide services to their users. They are characterised by high mobility and frequent topology changes, typically covering a wide geographical area. Lastly, an explanation of how mobility is addressed in ICNs was introduced while calling attention to some of the existing issues.

ICN design addresses many of the concerns that resulted in the traditional TCP/IP networks to be unfeasible for a mobility aware network environment. The way packet forwarding is handled is especially relevant due to the new way to which Data travels through the network, under the form of names. However, there still exist some open issues that are aim of discussion and which resulted in the appearance of many architectural branches under the ICN spectrum to this day, due to the different manners these issues can be tackled from.

Taking this into account, studies were made on how packet names could be advantageous to forwarding decisions in a mobility aware environment as well as the changes they imply in the Named Data Network forwarding workflow, culminating in the proposal of a context-based forwarding strategy in Chapter 3.

Chapter 3

Context-based Forwarding for Mobile ICN

After the description of the fundamental concepts addressed in this dissertation and along with a description of related work performed in these research areas, it is important to clarify what is the proposed solution in order to achieve an efficient Content discovery and forwarding process. Thus, in this chapter, the problem to be addressed is further detailed along with a forwarding mechanism proposal, while maximising its resolution by adjusting to network topology changes and packet Data.

This chapter is organized as follows:

- *Section 3.1*: *Problem Statement* - overviews the main problem that this work aims to address, which is the design of a Content discovery mechanism for a mobile ICN scenario;
- *Section 3.2*: *Macro-Architecture* - explains the NDN based macro-architecture and how the proposed changes will be reflect in its modules;
- *Section 3.3*: *Naming* - presents the Naming format choice and why it is essential to the discovery mechanism;
- *Section 3.4*: *Content Discovery and Forwarding* - describes the Content discovery mechanism proposed which takes into account context analysis and neighbourhood status;
- *Section 3.5*: *Forwarding Decision* - describes how the forwarding decision will be taken according to the type of packet;
- *Section 3.6*: *Final Considerations* - depicts the summary of this chapter.

3.1 Problem Statement

As previously mentioned, mobility introduces certain peculiarities in the network due to the ever-changing positioning of the nodes and their broad geographical dispersion, resulting in reduced contact time between nodes along with frequent network disruption. ICN tackles these issues by providing uniquely identified network Content to consumers by name-based routing rather than the routing of Data between a pair of devices [8]. In this way, ICN

presents native support for mobility, in a sense that its receiver-driven communication model allows consumers to request the Data they want by its identifier as well as the connection-less oriented method of delivering the requested information to a user.

Consequently, NDN supports this native mobility, from a consumer point of view, by allowing a mobile node to issue a new request for the Content whenever it moves out of reach of the previous provider and into a new point of connectivity. The main issue, therefore, lies with the ability to deliver the requested Content to the new location, in a form that is not noticeable to the consumer and, subsequently, which does not impact its experience. Moreover, publisher mobility represents a severe problem to solve since the continuity of a session needs to be maintained stable whenever a publisher is moving. However, there is no difference between the routing locator of a Content provider and the Content identifier [52].

The native mechanism used by NDN to transmit Content between consumers and publishers follows a best-route approach employing a least-cost metric. This proves to be inefficient because all traffic is treated in the same manner when it comes to forwarding, while Contents may belong to different context environments as well as possess various degrees of severity when it comes to loss of Data in the application layer, which is especially important for delay-sensitive scenarios. This means that, while NDN makes use of hierarchically structured names with some resemblance of human-readability, these are in no way considered in the forwarding workflow and the decisions taken, even though the potential to serve as auxiliary parameters exists. Moreover, the multipath capabilities of a node with different communication technologies are not fully utilized when they could increase the performance of the network in specific scenarios.

Furthermore, NDN makes use of Not-Acknowledged ($NACK$) packets to identify the occurrence of link failures, triggering updates on the network structures to forward future packets across different available paths, as well as mitigate some security issues such as flooding attacks. Compagno *et. al* [53] explore the usability of these packets both from a forwarding and security perspective in ICN, as well as their setbacks, and explain why they should be avoided. While this method is feasible for a continuously connected network topology, it cannot be applied to an ad-hoc scenario where wireless interfaces communicate in broadcast mode, as false positive NACK packets are bound to be sent due to the intermittent connectivy between nodes in the network. Thus, NACK packets are not considered for communications in the vehicular network between RSUs and OBUs in the scope of this work.

Within this context, this dissertation addresses these restrictions of NDN, building upon this concept by improving the way Content is discovered and forwarded in the network, while also taking into account the issues presented by node mobility, according to two key points: (i) analysis of the Content context to infer a proper baseline to the forwarding conditions required by that Data and (ii) study of the node's neighbourhood status to understand each of the node's interfaces conditions and provide multipath capabilities to the forwarding module. Moreover, while being a keystone of network design, security concerns were not addressed and consequently introduced in the evaluations, due to the focus being directed towards an effective delivery of Data to the nodes.

3.2 Macro Architecture

The proposed modifications to the ICN approach, while based in NDN, require some degree of separation between the functional modules to better understand their functionality,

and they are illustrated in Figure 3.1. These modules are *Interfaces & Mobility*, *Storage*, *Forwarding & Content Delivery*, and *Core* that includes *Naming*. All structures associated with the NDN design as well as the processes involved in the reception, analysis and routing of Interest and Data packets within a node are obviously included.

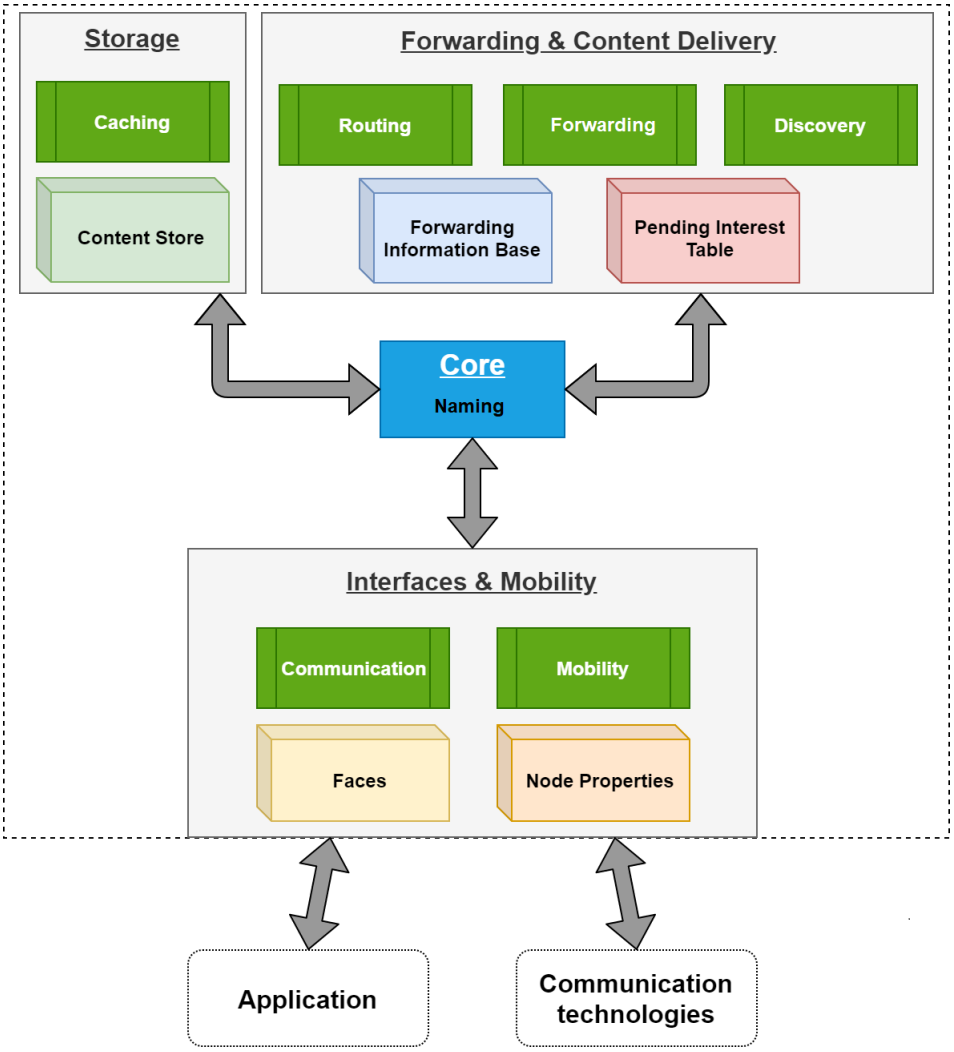


Figure 3.1: NDN-based macro-architecture overview.

The ***Core*** module interacts with all structures ensuring a logical and effective operation, and contains the Naming component which is transverse to all modules.

The ***Interfaces & Mobility*** module contains two processes known as Communication and Mobility. The first one takes charge of all the links between the device and the application layer, and with the outside, *i.e.* with other devices. The second one is related to the parameters of the node, that directly or indirectly influence its mobility. Two structures are assumed for the execution of these processes: *Faces* and *Node Properties*. *Faces* has the function of representing and managing the communication interfaces in the NDN stack, being an already established framework in the design; the *Node Properties* contains the inherent properties of the node, such as information regarding its mobility type, among others. The mobility

process is responsible for the mobile characterisation of any moving element, represented by a mobility profile or a neighbouring characterisation. This information is directly related to the node's properties, and it will be used by the remaining processes to take decisions on forwarding aspects.

The ***Storage*** module reflects the Caching process inherent to the NDN architecture, which has the function of storing Contents in memory to satisfy future requests. The *Content Store* (CS) is the responsible structure for storing Content, serving as the node cache. The purpose of this work is to improve the forwarding mechanism, in this way, the Storage module was not changed.

The ***Forwarding & Content Delivery*** **(F&CD)** assumes three components: Discovery, Forwarding, and Routing. The *Routing component* one is based on the use of a routing protocol to keep track of the paths available for each Content, which reflects in the periodic update of the network structures. In the context of this work, the well-known Named Data Link State Routing Protocol [54] was considered due to its established state and evolution alongside the NDN architecture. The *Forwarding component* relies on forwarding packets efficiently when available paths exist. It always takes collected information about the node's properties and packet Data before the decision. Lastly, the *Discovery component* differs from the previous two by serving as a complement to the routing mechanism whose goal is to determine the most rational path to forward Interest packets when there are no known paths available to reach the Content, in an effort to reach a provider first instead of directly discarding the Interest. The main information used by components of F&CD module is organised in two Data structures named Forwarding Information Base (FIB) and Pending Interest Table (PIT). The FIB structure maintains the information provided by the routing protocol, containing the entries with the name of each Content and their possible interfaces. The PIT structure maps requests waiting to be satisfied and the respective interfaces from which they came. The Forwarding component is the core of F&CD and houses strategies and optimisations aimed at improving NDN performance.

3.3 Naming

The structuring of Content names in a hierarchical manner is considered important for the scalability of the system and to formalise the required context for the Data to be processed. It takes into account that consumers reply to individual requirements of a large amount of Data. The use of name aggregation helps the search and sharing of Data because they have some degree of meaning to humans, and reflect some of the organisational structure of the Content. Due to the direct implications of the naming structure on the Content discovery mechanisms, a basic naming format will be used to complement our proposal. It is structured in five levels, as illustrated in Figure 3.2:

- **Application Category**, identifies which type of traffic the lower-level application will be associated with, e.g. Smart City Urban Planning;
- **Application Service**, identifies to which Content provider the Content belongs to, e.g. Traffic Management;
- **Application Domain**, identifies the Content domain and its relevance, e.g. emergency messages or feedback-based messages;

- **Content Context**, identifies the different naming information needed to infer its context concerning applications;
- **Content Attributes** is an auxiliary level to identify information specific to the Content, such as versioning or freshness.

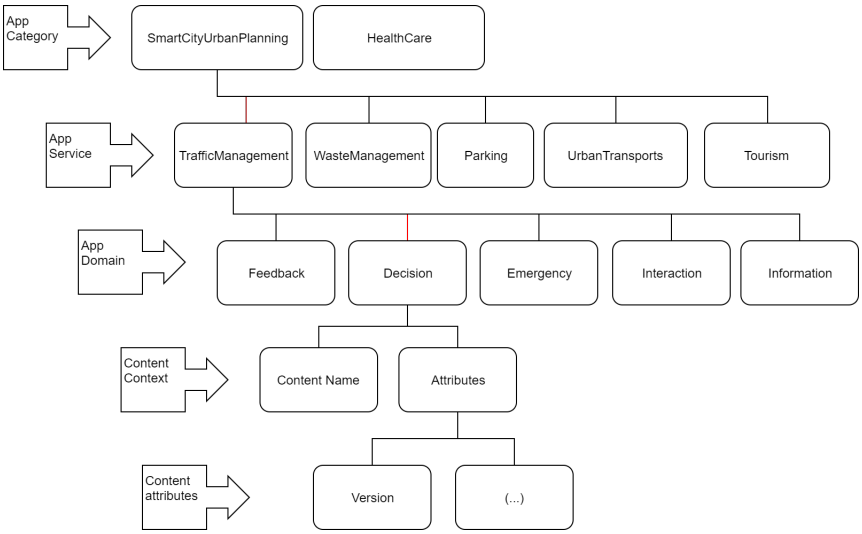


Figure 3.2: Proposed Naming Tree structure.

By using this model, one possible example of a resulting name for a Content being disseminated, under a Smart City context, would be:
//SmartCityUrbanPlanning/TrafficManagement/Decision/
"ContentContext"*/:::*"ContentAttributes"

The "/" sets the separation between components of the name in textual representation resembling how URLs are structured, having no meaning within the name, serving only for separation between the components. Moreover, the ":::" symbolises the separation between the name related to the Content and its attributes, which serve as additional information.

This information is encoded through a hash function when in a dissemination environment, for better space occupation and to facilitate and accelerate the search for records relating to the Content. This naming format will only make sense in the context of the applications and the Content provider. Thus, the proposed structure can be understood as the following:
//AppCategory/AppService/AppDomain/
"ContentContext"*/:::*"ContentAttributes" .

This structure allows the forwarding module to study part of the name included in the packets, with special emphasis in the application domain, gathering hints as to how important a given request is to the application layer and therefore serve as a weight in the following decisions to be taken.

3.4 Forwarding & Content Discovery

F&CD module considers all the processes related to the analysis and transmission of packets through the appropriate paths towards the Content. These can be split into two

specific phases: the uncovering of the Content in the network domain under the form of Interests packets, and the retrieval of such Content via Data packets.

In the primary NDN design (see Section 2.1), Content detection is achieved by consulting the FIB structure to identify possible paths for a specific Interest, and choosing one of them to redirect it, according to a forwarding strategy, which commonly follows a Least-Cost approach, choosing the interface which has the cheapest expenditure towards the producer. Retrieval is accomplished by forwarding the Data in the reverse path, using the PIT to identify through which interface the original request for that specific Content came from. While these approaches are tempting for a continuously connected network, this approach is unfeasible when in volatile conditions caused by wireless nodes and their mobility, such as constant changes of topology in VANET, since the cost and previous knowledge of the path taken by an Interest packet may become obsolete.

Forwarding should take advantage of the different communication technologies available on a mobile node, as well as to have a level of awareness of its neighbourhood state. Moreover, Naming, being one of the most important pieces of ICN concept, should also have an important role in the decision making, since by analysing the Content name of the packets, it is possible to infer the context as to which the information requested belongs (something which was not possible in IP networks). It is possible to study the relevance of such request in terms of priority or time-restrictions, and therefore, make decisions which are more appropriate to the nature of the request.

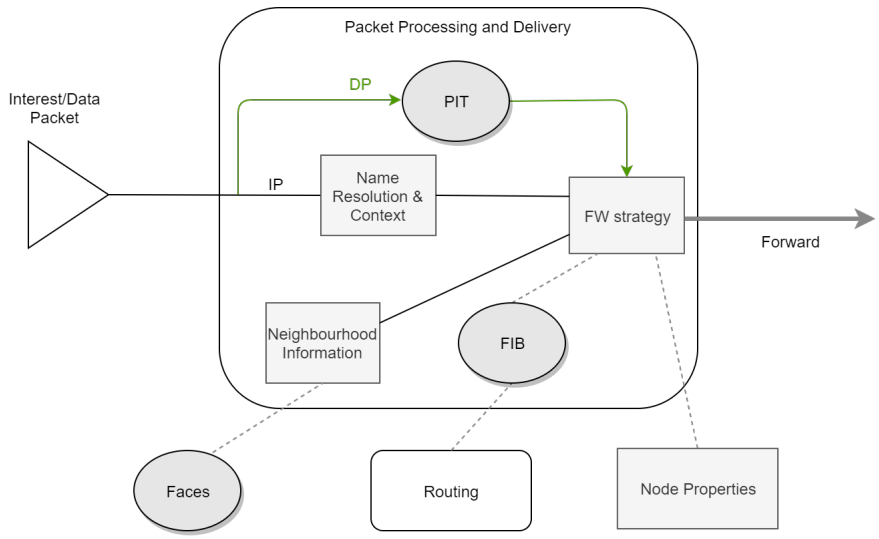


Figure 3.3: F&CD Flow.

Taking this into account, F&CD will be based in two different key aspects: Context Analysis and Neighbourhood Status. The general workflow of the considered changes are depicted in Figure 3.3 and will be detailed separately in the following sub sections.

3.4.1 Context Analysis

At this stage, a study and analysis of naming and packet Data is conducted to reflect the degree of importance of the Content being requested. To this end, three aspects are especially meaningful: *application domain*, *urgency* and *hop count*. These aspects are translated into

values, combined as shown in Figure 3.4, whose sum indicates the most likely successful strategy to use for packet forwarding (Figure 3.6).

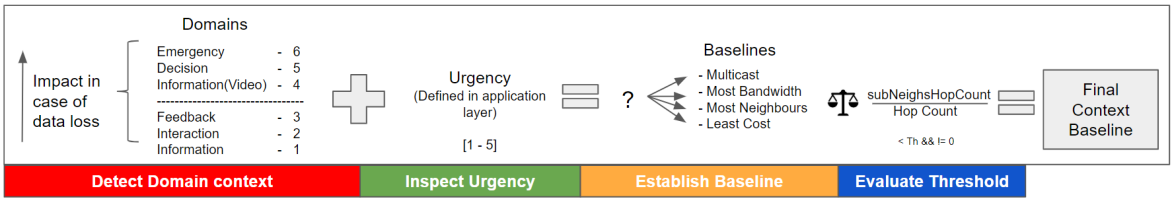


Figure 3.4: Context analysis workflow.

i) **Application Domain**

By taking advantage of human-readable hierarchical names inherent to NDN, it is possible to extract the domain to which the request belongs to in the broad spectrum of applications, and have a glimpse at its degree of importance (red slice in Figure 3.4). We split applications into six broad domains whose values and details are summarised in Figure 3.5:

- **Decision:** applications whose requests are important to make decisions and therefore should be delivered efficiently, also enclosing some degree of interactions between nodes, which are usually time-restricted. One such example is *traffic management*;
- **Feedback:** applications whose Data is used as notifications to some specific request. These applications are usually not time-restricted. One such case is *environmental monitoring by means of temperature sensors*;
- **Interaction:** applications which imply communication between nodes to infer state of some specific element. These are not time-restricted. One example is *car parking management*;
- **Emergency:** the most urgent applications that usually present a push-based nature in order to reach the majority of nodes, thus being time-restricted. One such example is *accident hazards*;
- **Information:** the most common applications, which represent sending and receiving Data. These can either be time-restricted (T-R) or not (No T-R), depending if they are video streams or simple files respectively. Due to presenting different conditions which are significant to an efficient delivery, each was attributed their own domain value (as Figure 3.5 illustrates).

Depending on the context analysis process under usage, these domains can be attributed values based on their time-constricted restrictions as well as the impact packet loss might have in the application layer towards the goal that Content is associated with. One such example is an emergency Content, in which, information is of the highest value since it can impact many users as well as have a high degree of severity.

	Constrictions	Domain	Value
Impact in case of data loss	Time-Restricted	Emergency	6
		Decision	5
		Information (Stream)	4
	Hybrid	Feedback	3
	Non Time-Restricted	Interaction	2
		Information (Data)	1

Figure 3.5: Context domains: details and values.

ii) **Urgency**

A parameter to help understand the seriousness and urgency of requests which, when used in conjunction with its context, indicates if it is advantageous to forward the packets through a more reliable connection (green slice in Figure 3.4). Within the context of this dissertation this attribute is defined at the application layer and is handled by the forwarding module without inferring its validity, which becomes a burden when applying this notion to a real-world network application. One proposal to reduce this issue, is to promote specific nodes within the network to a state where they can attribute urgency values to the Content requests based on their domains, in a pre-forwarding step, through an authentication mechanism, meaning that only trusted entities are allowed to forward packets with urgency values. In our proof-of-context we set the range from 1 to 5 for emergency values.

iii) **Subpar Neighbours links hop count**

By studying through which links a packet travels during its lifetime we can extract information regarding the nature of the neighbourhood of a given node. This is especially prevalent in the case that a specific packet has followed a particular forwarding pattern while still not having reached the Content provider. To this extent, a subpar neighbours hop counter was considered, standing for the number of hops a packet has travelled through interfaces which did not present the highest count of neighbour nodes, across all the available interfaces the node which transmitted the Interest had at the instant of that node's forwarding decision. This value can then be compared to the total number of hops the packet has roamed through. This way we can infer if an interface comprising the most neighbour nodes at that instant should be selected packet forwarding, according to an established threshold (blue slice in Figure 3.4).

The context analysis process starts by verifying in which domain the Content is inserted. At this stage, the forwarding mechanism acquires a glimpse of the level of attention that request should be given. However, the scope as to which Content belongs to is not enough to determine its necessity to be resolved on its own, as it may belong to a domain which in theory presents less value to the network, but poses very important Data to the entity requesting it, thus boosting its emphasis. Due to these concerns, the application domain is weighed

alongside the urgency level attributed to the packet in order to infer a baseline of which kind of connection could be more adequate for that specific context under specific fields, such as reaching the highest number of users possible or reaching the Content provider reliably (e.g. through the link with the most bandwidth). This baseline takes into consideration the type of links considered across the network and their impact in the congestion and bandwidth usage, as well as the possible values resulting from the domain and urgency analysis.

Taking into account that NDN wireless communications are performed in broadcast mode, these values were distributed equitably to each considered baseline types, *Multicast*, *Most Bandwidth*, *Most Neighbours* and *Least Cost*, giving less emphasis to the first and third ones in case those values are uneven. Furthermore, this distribution also contemplates an inclusive approach, in which baseline values can overlap across two different types, taking advantage of the urgency metric to serve as an arbiter to select in which type of link the selection should rest upon: if the Interest under analysis is assessed as urgent, a link of higher level is selected. Although this baseline could be sufficient to assume a primary forwarding decision, this approach deals with mobile nodes, and it can be further enriched by analysing the type of links that the packet has travelled so far: if the number of hops a packet has travelled through an interface with most neighbours greatly outnumbers the total hop count for a given Interest, under a certain threshold, and the Content remains to be found, the forwarder considers the choice of forwarding the packet via a link with a higher amount of neighbouring nodes if available, taking advantage of the conditions of this link in an attempt to give more opportunity to find the Content before the lifetime expires. This workflow is illustrated in Figure 3.4.

Taking into consideration the previously referred naming format, and the example there described, if a consumer sends an Interest for a Content, whose name's first components are *//SmartCityUrbanPlanning/TrafficManagement/Decision/*, the context baseline calculus would be the following: (i) the application domain, *Decision*, is extracted from the packet name with value 5; (ii) the established urgency value is retrieved from the packet tags, for this case in particular lets consider a value of 2; (iii) those two values are summed resulting in value of 7; (iv) the summed value is then compared to the set baseline values (as illustrated in figure 3.6), which would rest upon either a *Most Bandwidth* link or one with the highest amount of neighbouring nodes; (v) as the resulting value overlaps with two possible baselines, the retrieved urgency value is used as tie-breaker, choosing the link type with the *Most Neighbours*.

3.4.2 Neighbourhood Status

While the context analysis described in Section 3.4.1 focuses on the information carried by the packet in order to establish a forwarding baseline, it does not take into account the available interfaces and the proximity to other network elements, factors which can have direct impact on the choice of path to redirect the Content. Therefore, a thorough investigation of the node's neighbourhood is needed in order to complement that context baseline and give the forwarding strategy knowledge of the network conditions at that given moment. This information considers the network conditions concerning the forwarding node, evaluating the state of the neighbourhood with the available paths as listed in the FIB, to weight in the decision making according to different factors. Four aspects are relevant to infer the condition of a node's neighbourhood: available interfaces, number and type of neighbours, interface cost and the mobility pattern.

Domain	Urgency	Points Sum
Emergency (6)	[1 - 3]	7-9
	[4 - 5]	10-11
Decision (5)	[1 - 3]	6-8
	[4 - 5]	9-10
Information [T-R] (4)	[1 - 3]	5-7
	[4 - 5]	8-9
Feedback (3)	[1 - 3]	4-7
	[4 - 5]	7-8
Interaction (2)	[1 - 3]	3-5
	[4 - 5]	6-7
Information [No T-R] (1)	[1 - 3]	2-4
	[4 - 5]	5-6

$$\frac{\text{9 possible sum values}}{\text{4 forwarding baselines}} = \text{2.25 pts per baseline}$$

Urgency Inclusive Baselines

[10 - 11] Multicast
[7 - 9] Most Bandwidth
[5 - 7] Most Neighbours
[2 - 5] Least Cost

Urgency [4-5] Urgency [1-3]

Figure 3.6: Context baseline calculus, using urgency as overlap arbiter.

i) **Available Interfaces**

Nodes, while being mobile or static, will possess different communication technologies, commonly but not restricted to WAVE (802.11p), Wi-Fi (802.11n), and Ethernet. This factor is especially important in a mobile environment, where wireless conditions are under constant change, and choosing one specific technology might present more value than the rest.

ii) **Number & type of neighbours**

By investigating the number and mobility type (mobile or fixed) of neighbours per interface, a node can reliably understand the mobility conditions of its neighbourhood and, in conjunction with the analysis of the properties of a packet described previously, make the appropriate adjustments to the forwarding decisions.

iii) **Cost**

The cost of sending traffic through a certain interface is used as a unit of measure to determine the best route to reach a particular Content. Even though this work focuses on the context of named packets, the cost of an interface still continues to be a very important factor when performing decisions in conjunction with other metrics.

iv) **Mobility Input**

While the Mobility module is in charge of updating a node's neighbourhood parameters, such as their signal strength and mobility type, its input regarding the mobility pattern is also important since a node might infer future network conditions, such as future wireless connections based on prediction, usually by taking advantage of geographical attributes, such as heading and speed of a node.

3.4.3 Node Data Structures

In order to accommodate the new information to support the proposed forwarding heuristic, some new fields and tags are needed both in Interest and Data packets. Regarding the

Interest packet the new additions are three new tags: *HopCount*, which contains the number of hops; *SubparNeighsHopCount*, which contains the number of hops through links which did not represent the highest count of neighbours at the node in the instant the forwarding decision was taken; *Urgency* which indicates the urgency level of the request. As for the ***Data packet***, four new tags were added: *HopCount*, which contains the number of hops; *IntHopCount*, which indicates the number of hops of the Interest packet on that node; *SubparNeighsHopCount*, which contains the number of hops through links which did not represent the highest count of neighbours at the node in the instant the forwarding decision was taken; and *Urgency*, indicating the urgency level of the request. All these modifications are illustrated in Figure 3.7, marked in grey background color while original fields marked in white background color.

Lastly, the new structure in the ***Faces Table*** possesses three fields: *Type*, which indicates if the neighbour node is mobile or static; *SignalStrength*, which indicates the stability of the connection between both nodes at the time of the recorded information; and *Timestamp*, which indicates the time that entry was added to the Table. This addition is illustrated in Figure 3.8.

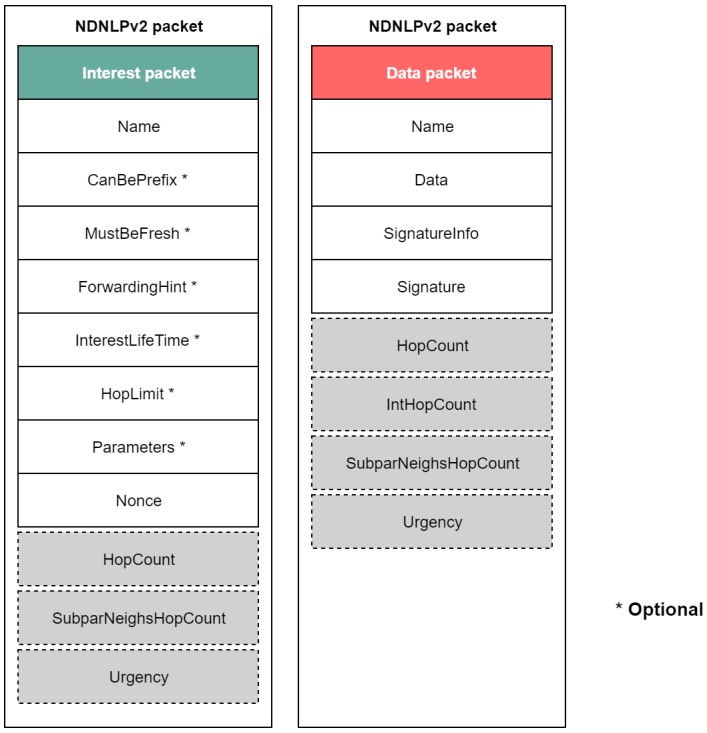


Figure 3.7: Communication packets format.

Face	LinkService	Transport	Counters	Cost	Neighs
1	...	...	...	2	...
2	...	...	...	7	...
3	...	...	...	6	...
4	...	...	...	3	...

Neigh	Type	SignalStrength	Timestamp
1	Mobile	57	1569456000
2	Static	-	1569456050
3	Mobile	89	1569456100

Figure 3.8: Faces table: new structure.

The neighbourhood analysis will be conducted straightforwardly according to each link type, by consulting the information that is updated periodically in the **Faces** structure, as seen in Figure 3.8. In the case of wireless links, the parameters to have in account are the number and mobility type of neighbours (chosen according to a signal strength above a certain threshold, along with a mobility pattern that is considered adequate compared to the node in question), as well as available bandwidth and link cost. In the case of wired links, the cost and bandwidth of the link are considered to be the primary parameters to have into account. With this information, its possible to counterweight the different available interfaces according to different conditions, thus allowing the creation of a list to be provided to the forwarding strategy, which identifies the interfaces most adequate to be used according to parameters such as most neighbours (*static/mobile/mixed*), least-cost or most bandwidth. An example of a resulting list after neighbourhood analysis occurs is illustrated in Figure 3.9.

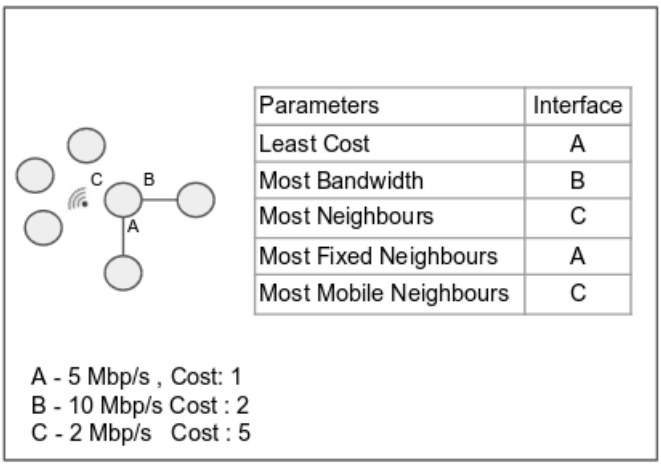


Figure 3.9: Example of a node's neighbourhood status retrieval.

3.5 Forwarding Decision

The two significant aspects described previously give crucial information regarding the conditions to which a specific packet belongs, and to agree on a compromise that has a good chance of successfully reaching the Content, even under dynamic conditions. This process is conducted by the ***Forwarding Strategy***, which acts as the central point of Data collection, obtaining not only the requirements of the Content request along with the current neighbourhood state, but also reap Data from the Data structures, FIB and PIT, and the properties of the node, as illustrated in Figure 3.3. The workflow of this decision process is directly influenced by the type of packet under analysis: *Interest* and *Data*.

3.5.1 Interest Packets

In case the packet under analysis is an *Interest* packet, its processing starts by searching the FIB structure. If there are one or more entries in the FIB for a specific name (symbolising there are known paths for that specific Content), then the neighbourhood analysis is instructed to only examine the interfaces associated with those entries. The resulting decision will

join the outcome of the context analysis concluding the Content discovery mechanism by forwarding the request through the most reliable path.

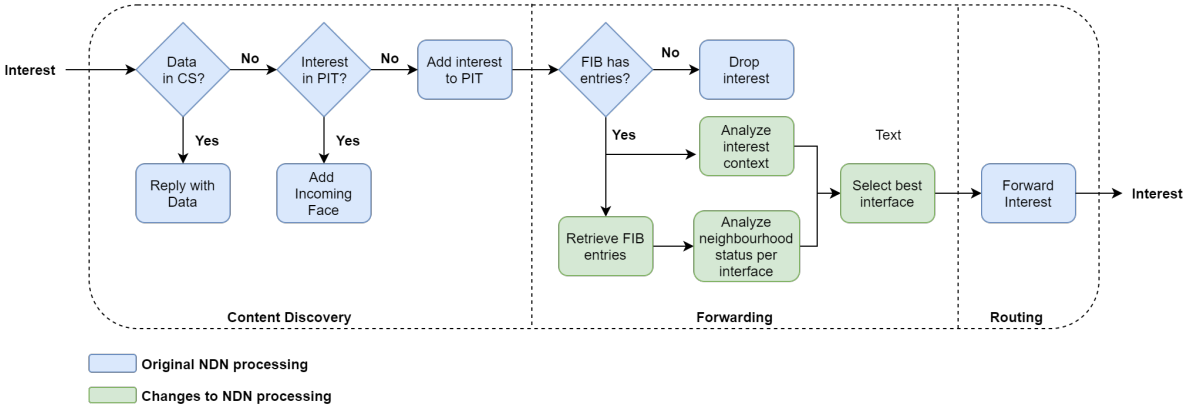


Figure 3.10: *Interest* packet forwarding packet decision process.

3.5.2 Data packets

In case the packet under analysis is of *Data* type, the process related to its treatment is similar to that of the original NDN concept, where the Content name is searched in the PIT to analyse if an entry already exists. In the positive case, the Content is forwarded in-reverse path through the interface from which the Interest packet came from. Additionally, in order to attend to mobility concerns, if a certain node receives a Data packet that was not destined to it and the variable *IntHopCount = 0*, two conclusions can be drawn: (i) it can be inferred that the consumer originated the Content request in the same neighbourhood of the current node; (ii) this means that the consumer may have moved recently and therefore, in a last attempt to make the Data reach him, the Data packet is diffused through all available interfaces.

It is also important to notice that NDN forwards the *Data* packet by the incoming interface of the *Interest* packet. However, when in mobile environments, this approach clearly limits the performance of the entire system due to the use of a broadcast mechanism in wireless interfaces, and the subsequently unnecessary Content transmission which might occur.

A possible solution to this flooding would be to include information in Data packets regarding a selective method which would trigger a mechanism in the forwarding module to identify if the node in question, which received the Data packet, should transmit the Data to its neighbourhood or remain silent. This approach would be similar to a clustering solution, in which certain nodes in a neighbourhood are selected to forward information as opposed to all nodes transmitting the Data, while not transmitting additional packets between themselves. However, this new information would have to be concise enough to allow all nodes to reach a conclusion based on their own properties without causing bottleneck in the Data delivery (*e.g. a node connected to the original consumer not being selected to forward the Data, causing the request to not be successful even though the Data was being provided*). Albeit raising knowledge of this concern, due to it not being the focus of this work, its study was not considered.

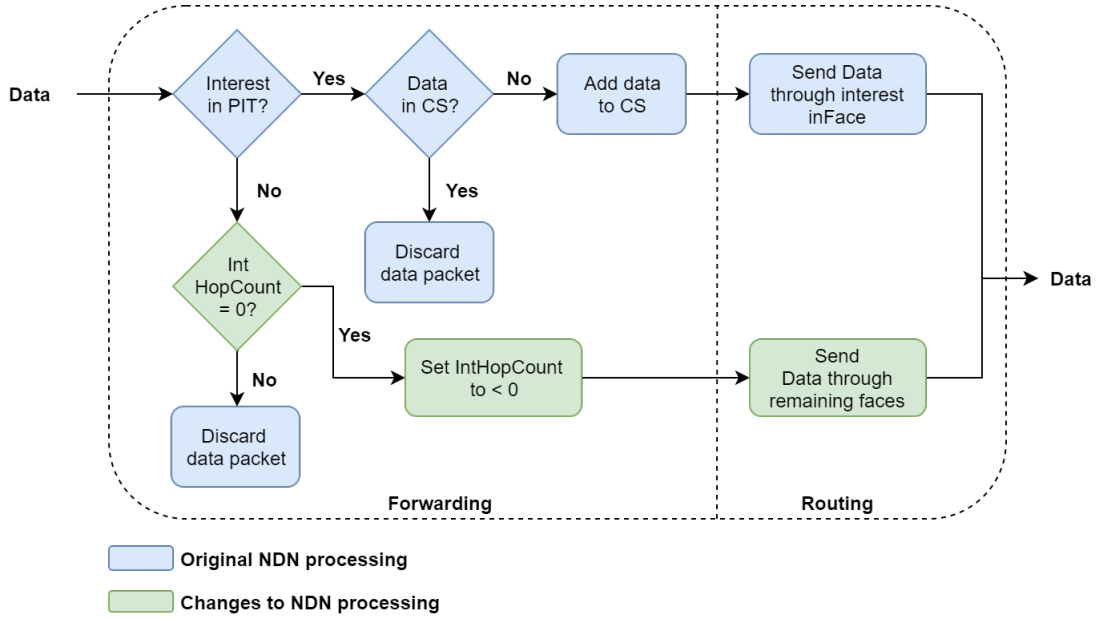


Figure 3.11: Data packet forwarding packet decision process.

3.6 Final Considerations

This chapter was focused on the description of the proposed context-based forwarding approach with all necessary adjustments and required knowledge. First an overview of the macro-architecture based on the NDN architecture was presented, followed by a suggestion of a naming format used in the context of this work to reflect the necessary adjustments. Subsequently, the forwarding mechanism proposed was presented, by describing the two different components necessary to forward information accordingly: (i) context analysis and (ii) neighbourhood status retrieval. Lastly, a workflow of the decision process was presented, describing how the forwarding strategy uses the parameters received to make a forwarding decision. The implementation of this mechanism is presented in the next chapter.

Chapter 4

Implementation and Integration

Once the Content discovery mechanism designed, it needs to be implemented. For that, the ndnSIM was the selected framework since it implements most of the features of the NDN communication model, presenting at the same time a simulation platform for evaluation tasks.

This chapter is organized as follows:

- *Section 4.1*: *ndnSIM Overview* - describes the main modules of ndnSIM as well as their functions;
- *Section 4.2*: *Drawbacks and Challenges* - discusses the drawbacks and hardships of implementing the proposed changes in ndnSIM;
- *Section 4.3*: *ndnSIM Changes* - explains the changes performed to the original ndnSIM software;
- *Section 4.5*: *Final Considerations* - summarises the chapter.

4.1 ndnSIM Overview

The ndnSIM[1] [6, 7] software is an *Open-Source Simulator Platform* of the well-known academic research-oriented network simulator, NS-3[2], which implements the NDN communication model, being specially optimised for simulation purposes. The ndnSIM implements all the basic features of the NDN architecture, including [55]:

- the hierarchical naming scheme and the Interest/Data packets format;
- the Data structures used for the packet processing, i.e., CS, PIT, and FIB;
- the basic Forwarding Strategy abstraction; and
- the Face abstraction, to support the interface between the NDN and the upper (applications) and lower (network, link, transport) layers.

The ndnSIM software allows [6]:

- to create simulation topologies (e.g. link bandwidth, node queue size, link delays, etc);

[1] https://ndnsim.net/current/index.html
[2] https://www.nsnam.org/

- to simulate available link-layer protocols models (e.g. point-to-point, wireless, etc);
- to simulate the exchange of NDN traffic among the simulated nodes; and
- to trace the exchange events.

The simulator is implemented in a modular fashion way, using separate C++ classes to model behaviour of each network-layer entity which comprises the NDN architecture: Data structures (PIT, FIB, and CS), network and application interfaces and forwarding strategy. This modular structure allows components to be modified at an individual level or replaced with minimal impact on other components. In addition, the simulator provides an extensive collection of interfaces and helpers to perform detailed tracing behaviour of every component, as well as NDN traffic flow.

The format of the *Interest* and *Data* packets used in the intervening communication between modules in ndnSIM are implemented based on ICN/NDN specifications presented in Section 2.1.4.5.

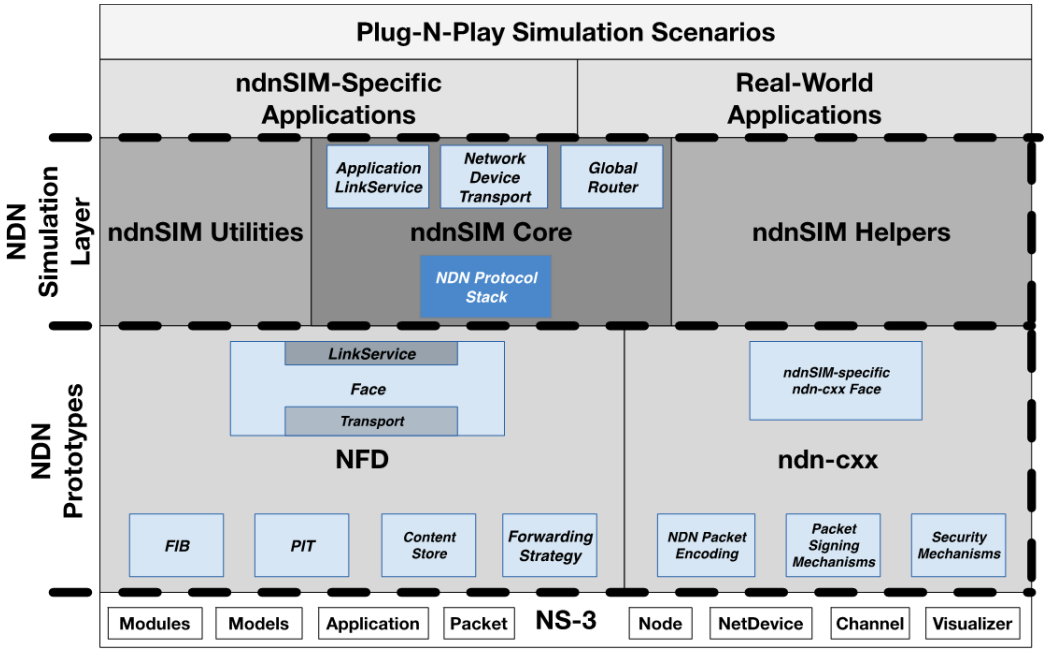


Figure 4.1: ndnSIM Ecosystem [6].

In order to better understand the ndnSIM Ecosystem (Figure 4.1 and its overall structure, a brief explanation of its main components and layers will be presented, consisting of the *Core NDN Protocol*, *NFD*, *ndn-cxx*, *Helpers*, *Utilities* and *Forwarding Strategy*.

4.1.1 Core NDN Protocol

The ndnSIM Core is responsible for implementing the NDN protocol stack and providing channel abstractions, allowing communication through the NS-3 ecosystem by means of Net-Devices. This module serves as the entry point for the decode and analysis of Interest and Data packets flowing through the network, as well as the retrieval of information from the FIB, PIT and CS structures to serve as directives for the choice undertaken by the Forwarding

Strategy on how to forward a particular packet. It also includes a global router module which eases the configuration of static FIB routes to quickly set up static network topologies.

4.1.2 NFD

The Named Data Networking Forwarding Daemon (NFD) is a network forwarder. It encompasses all the structures related to the NDN design, namely, FIB, PIT, CS, Forwarding Strategy, where information from the different modules culminate in a forwarding decision, and Face (Figure 3.8 in Section 3.4.2). The *Face* module also serves as a bridge between the application layer, in which applications determine how Interests are generated or Content is provided, and the NDN stack.

4.1.3 ndnSIM Helpers

ndnSIM makes use of helpers acting as points of abstraction for the software components in NFD, allowing their creation, configuration and management with ease. A set of helpers are also used to install both the NDN stack and simulated application parameters on NDN nodes for a simulation scenario.

4.1.4 ndnSIM Utilities

This module implements a number of packet tracers for simulation results from both the network and application layers as well as topology readers to simplify the definition of static simulation topologies, reducing the burden of implementing a logging system for every scenario.

4.1.5 ndn-cxx

This module relates to all the mechanisms associated with the treatment and preparation of packets to be managed in the NDN stack, by means of encoding and security signing and verifications. It is at this stage that all relevant NDNLVP2 packet tags need to be treated for proper sending and receiving between nodes. The ndn-cxx APIs are important for network-level, event scheduling, and absolute time operations.

4.1.6 Forwarding Strategy

The current ndnSIM software implements three simple Forwarding Strategies [56]:

- ***Best-Route strategy*** (default): If at least one outgoing face is available, the Interest will be sent through the one which presents the least cost. If none are available, the Interest is dropped, as there are no known paths for that Content.
- ***Flooding strategy***: Interests will be forwarded to all available faces registered in the FIB as having a route towards the Content, in broadcast mode. If there are no available faces, the Interest is dropped;
- ***Smart flooding strategy***: An extension of the flooding strategy, with introduction of a ranking system on faces which successfully deliver Content. If all available faces present the same ranking for a given Content name, the Interest will be forwarded across all, in

broadcast mode. If at least one face presents a higher rank for that Content, only that face will be selected for the sending of the Interest. Whenever a certain outgoing face is able to satisfy one Interest, the ranking of that face for that particular Content name increases, otherwise it decreases. If none faces are available, the Interest is dropped.

As referred previously, in Section 3.1, these forwarding mechanisms only take into account routing parameters, not being feasible for a dynamic environment where nodes are mobile.

4.2 Drawbacks and Challenges

While the ndnSIM software is an interesting implementation of the NDN protocol and has been used for community research, some noticeable concerns regarding its functionality were raised that impacted not only how it was perceived but also how some changes were needed to implement the proposal in this work.

- **Absence of routing protocol - NLSR**: ndnSIM in its current state does not include a full implementation of the Named Data Link-State routing protocol [54] which is in charge of updating the FIB structures with the available paths to reach any given Content regarding each node periodically as conditions change. In its absence, a global routing helper is used to install the network topology in each node at the start of simulation and it remains unchanged until the simulation finishes. However, this helper cannot be used to adjust the network topology over time. While NDN design natively supports mobility, which is further supported by NLSR, this absence means that the work developed, which addresses mobility concerns, cannot be tested in ndnSIM in its current state without performing changes. Moreover, implementing the NLSR protocol in ndnSIM would require serviceable time which is unfeasible considering the scope of this dissertation.

  Therefore, as an alternative, a multipath forwarding mechanism was implemented to support mobility by means of adjusting the FIB structure periodically. At the beginning of simulation, every node will have one single FIB entry resembling default Routes for all Contents (*e.g.* all Content can be reached from all available interfaces with equal cost) and the first time a Content is requested in a given node, the resulting Interest packet will be forwarded through all interfaces in a multicast manner. Then, for every Data packet that arrives, a new FIB entry is created or updated for that specific name with the cost and ID of the incoming interface, meaning that a path was found. This results in all future requests possessing at least a possible path to reach a specific Content. These entries are only removed after a Content request has expired or a NACK is received from the tried interface, meaning that the Content is no longer available. While at the start of simulation all Content requests will follow this process, after a path is registered in the FIB, all following transmissions will occur according to the forwarding mechanism based on context and neighbourhood as described in Section 3.4.

- **Restriction to downstream interface selection**

  Only a few works have used ndnSIM in wireless ad hoc networks. Furthermore, multi-hop wireless communication is not supported in the current official ndnSIM release natively, as restrictions exist to ensure a node does not consider the interface from

which the Interest was received in its forwarding decision, due to the fact that, in a continuously connected network, if the Content could be achieved from that interface, it would not have been forwarded in the first place. As a consequence, an Interest received from wireless interfaces cannot be straightforwardly forwarded over the same interface to permit multi-hop communication [55]. To this extent, adjustments had to be made to the NDN forwarder module in order to allow such communication to take place, by lifting the restrictions regarding the inclusion of the interface where the Interest packet arrived from, in case it is identified as an ad-hoc type.

- **Restrictions to Content size**: ndnSIM treats all Data as full files instead of chunks of a given Content, meaning that each name is associated with a single file. However, the simulator does not accept large sized files, below 2Mbs, meaning that simulations with high degree of segmentation in the network environment are not feasible without some adjustments. Our choice was to keep chunk as the base unit for Content.

- **FIB entries removed fully in case of Interest expiry**: As referred previously, a FIB entry is erased whenever a PIT entry expires or a NACK packet is received, meaning that the Content could not be found. However, it was found that whenever this occurred, the full entry would be removed instead of just making adjustments to which paths were chosen to forward the Interest. This means that the remaining available interfaces were being penalized while not being given a chance to deliver the Interest to a Content provider, which is specially noticeable in a mobility environment where disruptions are frequent. Thus, adjustments were made to the simulator in order to guarantee only the FIB next hops which did not receive a reply to a certain Interest are removed.

4.3 ndnSIM Changes

The discovery mechanism present in Section 3.4 requires some modifications in the ndnSIM architecture and protocol stack:

- **NDN Data structures**: In order to accommodate the new interface behaviour regarding the elimination of specific face entries in case of Interest expiry or NACK receipts, adjustments were made to the FIB structure. (Figure 4.2 green boxes).

- **Interest packet**: three new fields in NDNLPv2 packet to record the number of wireless hops (*wireless hop-count*), the number of total hops (*hop-count*) and the urgency (*urgency*) level of the request;

- **Data packet**: three new fields in NDNLPv2 packet to record the number of wireless (*wireless hop-count*) and total hops (*hop-count*) and the urgency (*urgency*) level of the request;(Figure 3.7) See details about the improvements in Section 3.4.3;

- **Faces**: addition of new field (*neighs*) to the existing structure, concerning a new table to reflect neighbour information, such as number and mobility type (*type*) of neighbours per interface, either static, mobile or a mixture or both, along with their respective received signal strength indicator(*signalstrength*), whose value fluctuates between 0 and 100, and UNIX timestamp (*timestamp*) of the instant when that information was registered, as depicted in Figure 3.8 and highlighted in blue in Figure 4.2.

The *LinkService* is the upper part of a Face, charged with the translation between network layer packets (Interests, Data, and Nacks) and link layer packets (TLV blocks), and bridges the gap between the desire of forwarding and the capabilities of the underlying transport. The *Transport* is the lower part of a Face and provides best-effort packet delivery service from and to the link service of the face, allowing connection to external sources. Lastly, the *Counters* table gives statistics about the count and size of Interests, Data, NACKs, and lower layer packets sent and received on the face.[3]

- **Forwarding Strategy**: the original best-route strategy is modified to use the inputs not only from the PIT and FIB Data structures but also from sources external to the packet processing module, highlighted in red in Figure 4.2, and detailed by Figure 4.3. This new information is the direct result from the analysis described in section 3.4 which makes use of packet context and neighbourhood status;

- **Forwarder**: The Forwarder class, as depicted in Figure 4.3, is the main mediator of all the processes occuring inside the NDN protocol stack, acting as the central point of information gathering and processing over the course of the forwarding procedure. Its workflow was changed to encompass the proposed modifications, by introducing Context Analysis and Neighbourhood Status retrieval, described in Sections 4.4.1 and 4.4.2, respectively;

- **NDN packet encoding**: small changes were made in the ndn-cxx component to allow for the correct encoding and decoding of the new tags introduced in the NDNvLP2 packet, as referred in section 3.4.3, when communication between Faces occurs in the transport layer; and

- **Mobility**: addition of an external module which is responsible for updating the faces table with information of a node's neighbourhood as well as mobility patterns, whenever a decision needs to be undertaken.

4.4 Implementation of Context-based Forwarding for Mobile

The following sections present the algorithms, procedures, and integration aspects about how the context-based forwarding strategies were implemented.

4.4.1 Context Analysis

The ndnSIM protocol stack (Figure 4.2) makes use of a Forwarder class that acts as the focal point of decisions on how to treat Interest, Data and NACK packets when they arrive at a NDN node. Thus, it is at this level that adjustments need to be made in order to study the context in which a particular Content is inserted and perform decisions accordingly. Due to the Data packets being addressed in a similar to that of the original NDN concept design, this procedure will mostly address Interest packets in an attempt to treat them in a manner this is preferable to the Data being requested. The Context Analysis procedure is formalised in Algorithm 1.

[3]NDN, Technical Report NDN-0021 http://named-data.net/publications/techreports/

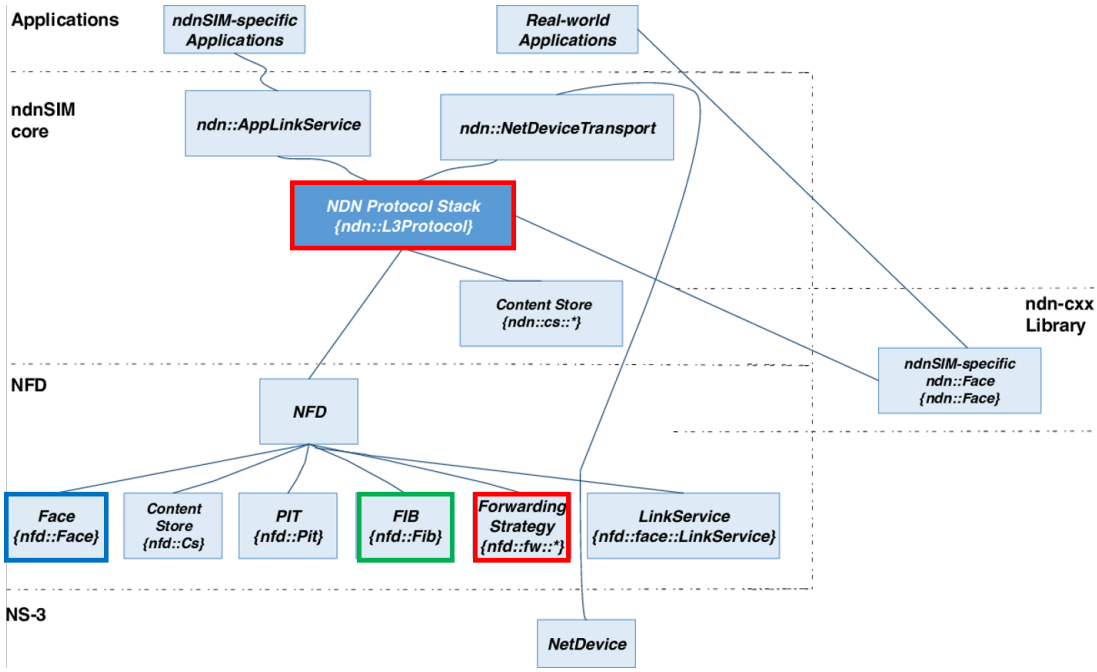


Figure 4.2: Structural diagram of the ndnSIM design components [7].

Algorithm 1 Content Context Analysis

```
1:  procedure ContextAnalysis
2:      Retrieve domain component from received name
3:      if domain is VALID then
4:          Set value according to domain impact
5:      Retrieve Urgency value from received packet tags
6:      if urgency value is VALID then
7:          Add urgency value to context value
8:      if sum >= minMulticastValue then
9:          baseline = Multicast
10:     else if sum >= minBandwidthValue + 1 or sum == minBandwidthValue or urgency >= 4 then
11:         baseline = MostBandwidth
12:     else if sum >= minNeighboursValue + 1 or sum == minNeighboursValue or urgency >= 4 then
13:         baseline = MostNeighs
14:     else
15:         baseline = LeastCost
16:     Retrieve Wireless Hop Count from packet tags
17:     Retrieve Total Hop Count from packet tags
18:     if subparNeighsHc / Hc <= Th and subparNeighsHc / Hc != 0 then
19:         baseline = MostNeighs
20:     return context baseline
```

This procedure is initialised by retrieving the domain component from the name that is associated in the packet (line 2). If this attribute is valid, considering the possible domains presented in 3.4.1, then a value is attributed according to the impact that domain induces in the application domain in case of Data loss as well as the time-restrictions it can include (line 4) otherwise the packet is considered invalid and, thus, discarded. Next the *Urgency* level is retrieved from the packet tags (line 5) and, in case it is valid, is added to the previous defined domain value (line 7). Taking into account that the range of possible values resulting from this addition is known, by splitting it with the four previously defined values for Content context,

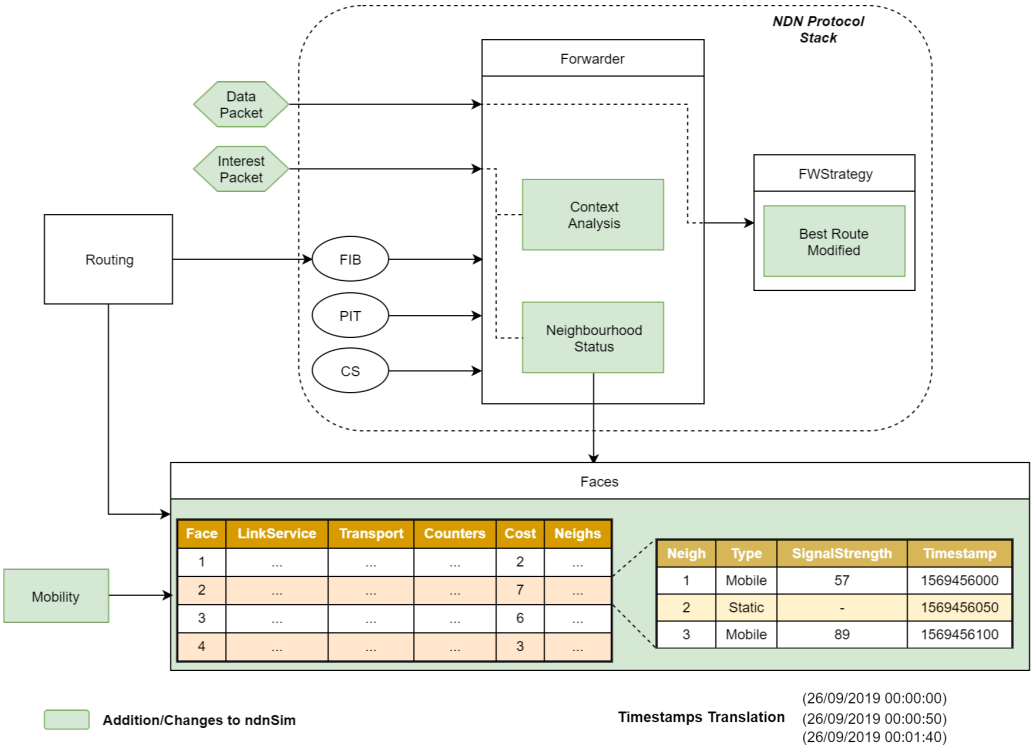


Figure 4.3: Changes to Forwarding module in ndnSIM.

a distribution of values is performed and attributed to each possible context value. Since these values are not equally distributed, urgency is also taken into account and baselines can be defined to different contexts by analysing if a Content was urgent or not. Thus, a context baseline is defined according to the sum between the urgency and domain values (lines 8–15).

Last but not least, due to the nature of a multi-path approach, it is important to keep tabs of the type of connections a certain Interest packet as travelled in order to weigh in if adjustment to the forwarding decisions should be made. The values of Subpar Neighs hop counts and total hop counts are then retrieved from the packet tags to be analysed (lines 16–17). If the average number of hop counts the Interest has travelled through links which did not represent the highest count of neighbours at the node at the instant the forwarding decision was taken, is below a defined threshold then the context baseline defined previously is adjusted to reach the highest number of neighbours available in an attempt to reach a Content provider (lines 18–19). At this stage, the final context baseline is defined.

4.4.2 Neighbourhood Analysis

In order to take advantage of neighbourhood status, the Forwarder module was updated so that a vector map can be retrieved by analysing the *Faces structure* (Figure 4.3), and inform the forwarding strategy of the network conditions a node is in at the time of packet processing. The Neighbourhood Analysis procedure is formalised in Algorithm 2.

This procedure begins by creating a vector map with FaceId values for all possible interface attributes. This map will be updated throughout the neighbourhood analysis as each node interface's condition is studied. Since the running routing protocol has the responsibility of

Algorithm 2 Neighbourhood Analysis

```
 1: procedure NeighStatus
 2:     Create FaceId Map with each possible type
 3:     Get and Insert next FIB hops of received Name into hops list
 4:     for face = firstFace to face = lastFace do
 5:         if face in hops list and scope == NON_LOCAL then
 6:             if face[NeighsNum] >map[NeighsNum] then
 7:                 map[NeighsNum FaceId] = faceId
 8:             if face[MobileNeighsNum] >map[MobileNeighsNum] then
 9:                 map[MobileNeighsNum FaceId] = faceId
10:             if face[FixedNeighsNum] >map[FixedNeighsNum] then
11:                 map[FixedNeighsNum FaceId] = faceId
12:             if face[Cost] >map[Cost] then
13:                 map[Cost] = faceId
14:             if face[Bandwidth] >map[Bandwidth] then
15:                 map[Bandwidth] = faceId
16:     return FaceId map
```

updating the FIB structure with the available paths to reach a Content provider for any given announced name prefix, it only makes sense to study the neighbourhood conditions of the faces which have valid next hops towards the name under processing. Thus, a list of valid next hops is created by checking the FIB entries for the received name prefix (line 3). Then, for each face the node in the hops list, a series of scannings are performed regarding the total number of neighbours that interface can connect to, along with their mobility type, the interface cost and the available bandwidth. In the case any of these values is higher than the one currently registered in the vector map, this value is adjusted (lines 6-15).

At this stage, both the packet name context and the node in question's neighbourhood status were analyzed and are ready to be consumed by the forwarding strategy in order to select an interface to forward the Interest. The procedure of how these parameters will be used to come to a decision are described in the following section.

4.4.3 Forwarding Strategy

The original strategy assumed a best-route stance, forwarding Content through the interface with the least routing cost as the sole metric in the decision. From this basic algorithm we made the necessary changes and increments for forwarding to consider context analysis and neighbourhood status in deciding which path the packet should follow. As has been extensively discussed in previous sessions, forwarding should handle *Interest packets* and *Data packets*. How each of them present distinct requirements they should be receive distinct procedures. The following are the algorithms and procedures for each packet type.

4.4.3.1 Interest packet

The Best Route Modified procedure for *Interest packets* is formalised in Algorithm 3.

Through algorithms 1 and 2, a retrieval of both the context analysis (line 2) and the node's neighbourhood status(line 3) is performed. Then, it is only feasible to study the available next hops as defined in the FIB entry for the name prefix under analysis, due to those being the only paths towards the Content providers (line 4). For each of the considered next hops, an evaluation is performed regarded the type of link they are associated to (line 6). In case the registered type is not ad-hoc and the ID of the next hop being analyzed is equal to the

Algorithm 3 Best Route Modified – ***Interest Packets***

```
 1: procedure SendInterest
 2:     Retrieve context baseline from analysis
 3:     Retrieve neighbourhood status map from analysis
 4:     for hop = firstNextHop to hop = lastNextHop do
 5:         Retrieve hop Face structure
 6:         if face Id == inFace Id and face link type != Ad-hoc then
 7:             Would Violate Scope
 8:             continue for loop
 9:         if face Id exists in neigh map for context baseline then
10:             Send Interest through face
11:             set Sent tag to true
12:             break for loop
13:     if Sent tag != true then
14:         Send Interest through Least Cost interface
```

ID of the face where the Interest arrived from, the original restrictions to wired interfaces still apply and, since a violation of scope occurs, the next hop is no longer considered for that particular analysis (line 7). Then, a comparison if performed between the current faceId under analysis and the context baseline defined. If this faceId exists in the vector map for the context defined (line 9), it means that this packet requires special forwarding conditions and is therefore sent through the corresponding interface (line 10), triggering a flag which indicates the Content was forwarded successfully. Finally, if no decision was taken (line 13) after analysing through all next hops, this means the context analysis was inconclusive and the Interest is forwarded through the least cost interface (line 14), as per original NDN design.

4.4.3.2 Data packet

It was referred in the Section 3.4 that the forwarding processing for Data packet would remain similar to the original, with the exception of addressing the event when a consumer requests a certain Content and then moves to a different location. This introduced a higher number of packets travelling in the network due to the re-transmission process for the same Content not acquired at the previous location. This work attempts to tackle this concern by resorting to a last effort mechanism in which a node verifies if the original consumer was nearby at the time the Content request was originated and diffuses the received Data through all interfaces in an effort to reach the consumer. In this way, the Best Route Modified procedure for *Data packets* is formalised in Algorithm 4.

Algorithm 4 Best Route Modified – ***Data Packets***

```
 1: procedure SendData
 2:     Retrieve Interest Hop Count from packet tags
 3:     if intHopCount == 0 then
 4:         Set Data packet's intHopCount to 0
 5:         if inFace link type == ad-hoc then
 6:             Send Data packet through all available interfaces
 7:         else
 8:             Send Data packet through all remaining faces
 9:     else
10:         Retrieve Interest inFace entry from PIT
11:         Decrement Data packet's intHopCount value
12:         Send Data packet through Interest inFace
```

This procedure begins by retrieving the attribute *IntHopCount* from the packet tags (line 2). If this value is equal to zero (line 3), the node assumes the consumer was in its neigh-

bourhood at the time the Content was requested, updating the *IntHopCount* attribute to a negative value (line 4) and sending the Data one last time with two possibilities:

(i) through all available interfaces (line 6) in case the link type of the incoming interface was ad-hoc or

(ii) through all remaining interfaces otherwise (line 8).

In case the attribute *IntHopCount* is not equal to zero, this means the Data is still under dissemination and therefore the corresponding Interest incoming face is retrieved from the PIT (line 10) reducing the value of the attribute by one (line 11).

4.5 Final Considerations

This chapter focused on the changes performed in ndnSIM for the implementation of the forwarding mechanism here proposed. Although the software is an efficient tool to design and quickly evaluate both caching and forwarding strategies, the built-in code was not prepared for scenarios enveloping mobility. A description was first covered concerning the modules built into the software as well as their functionalities. Then the hardships encountered while developing the simulator were also raised and solutions were proposed to minimise their impact in the evaluation of the forwarding mechanism presented. Finally, the required changes to the existing modules were detailed and pseudo-code versions of the changes how forwarding works in NDN were introduced to further explain how the implementation will work in the simulation environment.

Chapter 5

Tests and Results

After the implementation and integration of the proposed strategy an evaluation process is mandatory in order to understand its performance compared to previous implementations. Thus, this chapter describes the application scenarios and Use Cases considered, as well as the results obtained.

This chapter is organized as follows:

- *Section 5.1*: *Application Scenario and Datasets* - presents a detailed description of the Use Cases evaluated, covering a set of parameters as number of nodes, geographical area, dissemination periods, among others;
- *Section 5.2*: *Evaluated Metrics* - explains the choice of metrics and why they matter to the evaluation of the discovery mechanism;
- *Section 5.3*: *Results* - explains the results obtained; and
- *Section 5.4*: *Final Considerations* - summarises this chapter.

5.1 Application Scenarios and Datasets

The purpose of the realized tests is to evaluate the efficiency of the context-based forwarding strategy and the effectiveness of Content delivery in the several configurations. Considering the several types of Contents that can flow through the network, the analysis will consider two distinct Use Cases. First, where only urgent Content is available, in order to understand if the proposed strategy improves the delivery of Content in critical environments, where the number of available Contents is low but relevant to the users which consume it. Then, we will consider a second Use Case where both urgent and non-urgent Contents co-exist, symbolising a more natural environment, where concerns such as congestion and Content redundancy exist. To this extent, the evaluation was carried out by studying its performance in accordance to mobility Data retrieved from a real testbed in Porto city.

5.1.1 Application Scenario

A network following an ICN approach rests on the notion of forwarding packets by their names, whose definition is of the application layer. Therefore, it is comprised of a wide variety of Content with different characteristics, which means that the variables involved in

the analysis of the proposed mechanism are many. To this extent, this work will focus on file dissemination under a Touristic scenario, in which Data providers are fixed nodes (RSUs) and the ones consuming that Data are mobile (OBUs). The goal set here is that end-users circulating through a smart city while connected to the network may request *non-urgent Content* (Use Case 2 – Section 5.3.2) related to their neighbourhood, such as restaurant suggestions or cultural spot descriptions while on the move, as well as *urgent Content* (Use Case 1 – Section 5.3.1) such as emergency broadcast messages, like environmental hazards or traffic accidents.

Figure 5.1: Touristic Application Scenario.

5.1.2 Porto Dataset

A vehicular network located in Porto city has been deployed in common projects with both Universities of Aveiro and Porto, IT, and Veniam, and is supported by an infrastructure network of Porto Digital [57]. This testbed interconnects hundreds of vehicles, such as public buses, garbage trucks and municipal vehicles, in order to provide a set of services such as Internet access and delay-tolerant communications to transport non-urgent information (sensoring and logs). A Data collection of the network behaviour was performed, comprising a period of 4-hours from 9 am to 13 pm which includes both Rush-hour and non Rush hour periods, as described in Table 5.1. This Data is composed by: nodeID, heading, GPS position (longitude and latitude), speed of the node, and GPS time; per time sample.

Period		Region	OBUs	RSUs	Backend Routers
Rush Hour	09am-10am	City Center	80	16	4
non-Rush Hour	10am-01pm				

Table 5.1: Number and type of nodes evaluated.

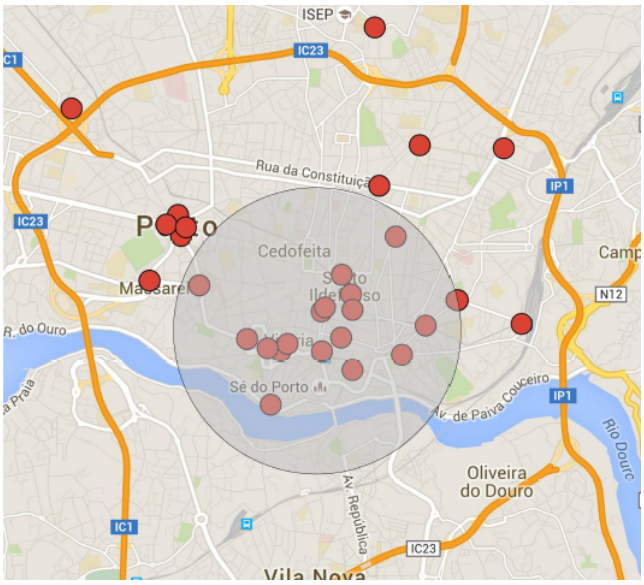


Figure 5.2: Porto Dataset geographical location (vehicles as red circles).

5.2 Evaluated Metrics

Three main metrics were selected to measure the proposed forwarding strategy: satisfied Interests, transmission delay and network overhead. The first metric represents a measure of consumer effectiveness, and the two others, assess efficiency from a network standpoint. In addition, we refine each of them by evaluating specific aspects whose details are given below.

5.2.1 Satisfied Interests rate

In order to understand how the proposed context-aware forwarding strategy improves the existing one, it is important to study how the forwarding module addressed the Interest packets, both from a consumer and a global network point of view. To this into account, the following related metrics were considered relevant:

i) **Consumer Satisfied Interests** is considered the number of Content requests which successfully got a Data reply. This is the primary metric to keep in mind as it indicates that a user got the information he requested.

ii) **Consumer Timed Out Interests** is considered the number of Interest Packets originated by consumers which never got an answer to a sent Interest, ending up expiring due to the lifetime running out in the PIT. If a certain Content is in high demand, a higher count of timed out packets will mean more re-transmissions need to occur for the same Content in an effort to retrieve the Data.

iii) **Satisfied Interest** is the number of Content requests which successfully got resolved, by means of a Data packet retrieval, on a global network scale. It differs from the number of consumer satisfied Interests due to the fact NDN forwarding works on a hop-by-hop basis and, therefore, all intermediate nodes (nodes which did not originate a Content request) will both send and receive Interest packets.

iv) **Timed Out Interests** is the number of Interest packets expired without receiving a Data reply, on a global network scale.

v) **Cache Hits** is the number of Interests that successfully arrived at a node which possessed the Content requested, thus being able to provide it in return. It serves as metric to understand the degree of how many Interests actually reached the Content, knowing that the Data may never reach the consumer mobility and link failures.

5.2.2 Transmission Delay

Although the goal of the proposed strategy is to increase the availability of Content in the network, it is important to do so in a manner that does not impact negatively the delays. It is measured from the instant when the consumer originates an Interest up until the instant a Content is discovered and the Data returns in reverse path. Keeping this in mind, the following metrics were considered relevant:

i) **Full Transmission Delay** is considered the time occurred between the first Interest packet created for a given Content, up until the time the Data is received. This metric takes into account that Interest expiry occurs and thus includes all the required re-transmissions.

ii) **Last Transmission Delay** is considered the time occurred between the sending of an Interest packet created for a given Content, up until the time the Data is received for that same Interest, not accounting for re-transmissions. Thus, only Interests receiving Data replies are considered.

5.2.3 Network Overhead

Similarly to the previous notion of maintaining transmissions delays stable across the network lifetime, it is important to study the overhead status which may have been introduced by the proposed forwarding strategy changes. It is also important to notice that overhead escalates in an ad-hoc network and can lead to congestion, thus impacting directly the network performance. The following metrics were considered:

i) **In Interests** is the number of Interests received by nodes on a global network scale. This value is affected before any Content name validation is performed, meaning that Interests received can still be dropped later, due to a number of reasons.

ii) **Out Interests** is the number of Interests sent by nodes on a global network scale.

iii) **In Data** is the Data packets equivalent to the "In Interests" metric. This value is impacted by whenever a Data packet is received and before any validations are performed, meaning that it can be unsolicited and, thus, eventually dropped.

iv) **Out Data** is the Data packets equivalent to the "Out Interests" metric.

v) **Hop Count Sum** is the number of hop counts travelled by an Interest packet in the network in order to reach the respective Content provider, starting from a consumer node.

vi) **Content Re-transmissions** is considered the number of Interest re-transmissions needed for a certain Content to be received at a consumer node.

5.3 Results

Next we discuss the results obtained by our strategy for Use Case 1, urgent requests, and for Use Case 2, mixed requests.

Initially, a comparison will be made between the forwarding strategy inherent to NDN design, which will be referred to as "native", and the changes proposed in this work, to which we will refer as "context-based". Two Use Cases will be considered, the second being an evolution of the first one, with added complexity. The configurations which were considered and installed in each node for the experiments are described in Table 5.2. It is that assumed mobile nodes have limited cache sizes (in number of Data Packets), so many replacements will have to occur over the course of the simulation.

	OBUs		RSUs	Routers
Node Type	Consumers	Intermediate	Producer	Backend
Installed Technologies	WAVE WIFI		WAVE WIFI Ethernet	Ethernet
Mobility Type	Mobile		Fixed	
Cache Size	20	20	60	100
Number	40	40	16	4

Table 5.2: Node configurations for evaluated Use Cases.

Additionally, two variants were considered for both Use Cases regarding the distribution of Content across all providers. First, a case where all Content providers are able to address requests for all files. This variant will be denominated as "**Equal**". Second, a case where Content files are split evenly across producers, meaning that in the best case event, a producer will only be able to answer to half of the Interests locally, and will have to look for the remaining half via forwarding, either through the back-end routers or through its wireless neighbourhood. This second variant will be referred to as "**Diff**". Lastly, the ndnSIM software implements two native caching policies, Least-Recently-Used (LRU) and Priority First-In-First-Out (PFIFO) [58], both being considered in the forwarding strategy evaluation, in order to study which is better suited for the changes proposed.

5.3.1 Use Case 1 - Urgent Requests

This Use Case focuses solely on the dissemination of Content which is considered *Urgent* (presents maximum value of "**Urgency**" in context analysis). The considered parameters are presented in Table 5.3. All application domains were considered since each may induce different forwarding decisions according to their impact in the application layer (see Section

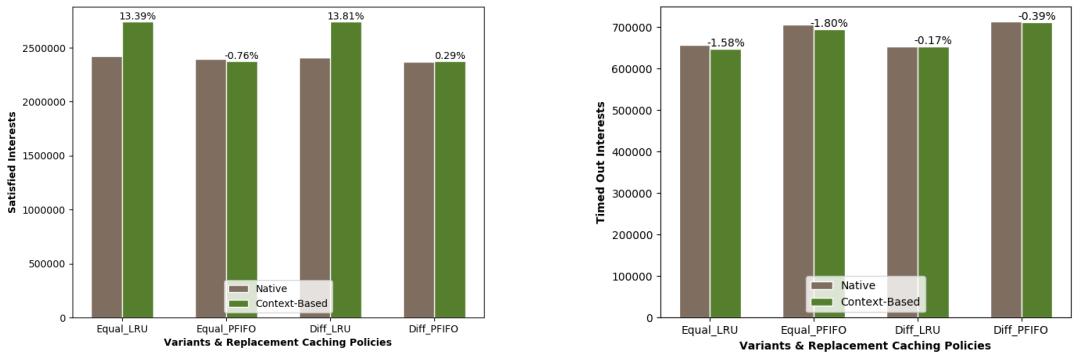
3.4.1). For visibility, both information domains were aggregated into a single one in the table. On one hand, consumer nodes will request all available Contents regarding each domain, with maximum urgent value and at a frequency of one Interest per second. On the other hand, producer nodes will provide five files for each Content with increasing freshness values, due to the consideration that more impactful domains might have higher volatility when it comes to Data, in a similar fashion to emergency messages.

Content	Consumer (Zipf Distribution)			Producer	
	Requests	Urgency	Rate (int/sec)	Available Content	Freshness (seconds)
Emergency	All	Urgent	1.0	5	900
Decision					1800
Feedback					3000
Interaction					3000
Information				10	3600

Table 5.3: Parameters used in Use Case 1.

5.3.1.1 Satisfied Interests Rate

Figure 5.3(a) presents the results obtained regarding the satisfied Interests across all nodes in the network, including consumers, intermediate nodes and producers, while Figure 5.3(b) presents the number of Interests which timed out, due to lifetime expiry.



(a) Satisfied Interests.

(b) Timed Out Interests.

Figure 5.3: Use Case 1 - Global Interest overview.

The results show that the context-based implementation increased Interest satisfaction around of 13% when using the LRU caching policy over the native one, in both the equal and different cache variants. However, for the PFIFO caching policy performance decreases slightly. Since the rate of Interests satisfied increases, as well as the number of cache Hits on a global scale, as suggested in Figure 5.4, it is expected that a lesser amount of Interests will expire.

Figure 5.4 depicts the number of cache hits occurred in the network over the course of the simulation. It is possible to verify that the context-based implementation allows for a

higher number of Interests to successfully reach a Content provider, being more significant when using the LRU caching policy. This value demonstrates how the forwarding module takes advantage of connection links which may not necessarily follow the least cost metric, but may provide better Content availability by means of better stability.

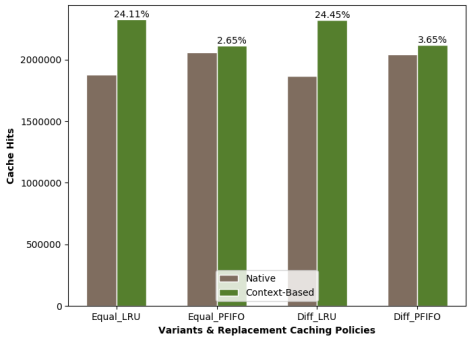
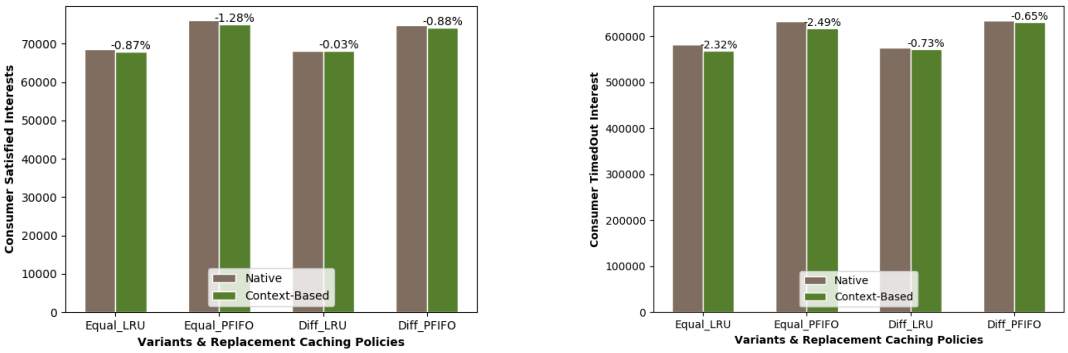


Figure 5.4: Use Case 1 - Global cache Hits overview.

While the previous Interest satisfaction rates follow a global perspective, they do not provide enough information to infer if consumers, which originated the requests, are actually receiving the Content in return. To this extent, Figures 5.5(a) and 5.5(b) illustrate both these metrics, focusing solely on consumer nodes. In this stage, a conclusion can be drawn that the context-based strategy displays similar results, even though the global Interest satisfy rate increases, as illustrated in Figure 5.3(a). An explanation for this behaviour rests in the fact that a consumer is constantly on the move, meaning that the corresponding Data packet might never make it back successfully before the Interest expires, as this procedure is done in-reverse path.



(a) Satisfied Interests.

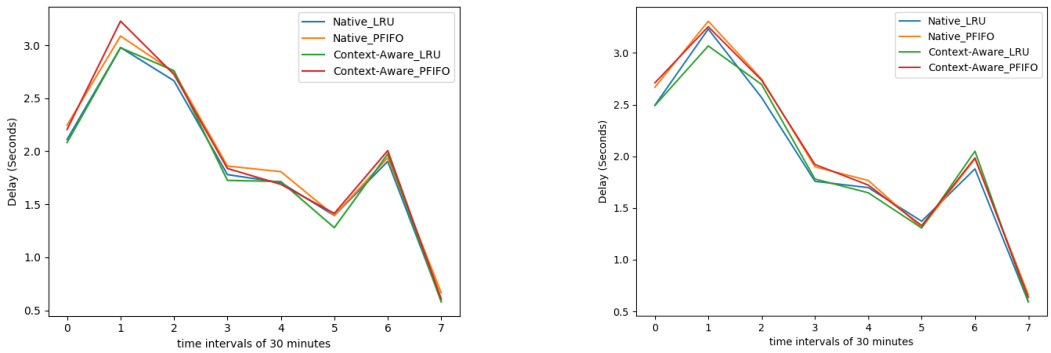
(b) Timed Out Interests.

Figure 5.5: Use Case 1 - Consumer specific Interest overview.

5.3.1.2 Transmission Delay

Figures 5.6(a) and 5.6(b) illustrate the full delay between the transmission of an Interest and the reception of the correspondent Data packet, including Interest re-transmissions which

might have been needed, over the course of four hours and split across thirty minutes intervals. As values were registered only after each of these time frames, the first value depicted in both graphs corresponds to the first thirty minutes of simulation. The context-based strategy with LRU caching policy shows slightly lower performance only in the third half hour (interval 2) of simulation. The rest of the time, context-based satisfies Interests equally or more efficiently than other strategies. Likewise, the PFIFO caching policy shows lower performance in the third half hour of simulation when the Equal cache variant is considered, while behaving equally or more efficiently in the remaining time.



(a) Equal Cache Full Delay.

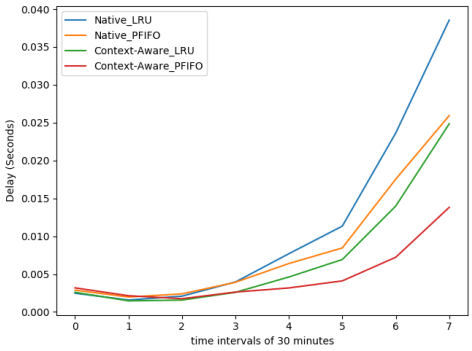
(b) Diff Cache Full Delay.

Figure 5.6: Use Case 1 - Full Delay study for both Equal and Diff variants, following a thirty minute interval observation period.

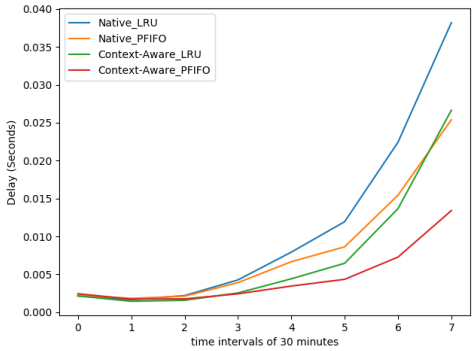
The last delay metric discards all notion of re-transmissions and focuses solely on the time needed for an Interest packet to reach a Content provider and trigger a successful Data delivery procedure. Figures 5.7(a) and 5.7(b) portray this, over the course of the full simulation and by studying intervals of thirty minutes. In this regard, a conclusion can be drawn that the context-based strategy achieves higher performance than its counterpart, which can be explained by the fact that as more Interests successfully discover a Content provider, as seen in Figures 5.3(a) and 5.4, the Content will be available closer to the nodes requesting it, since the Use Case follows a Leave-Copy-Everywhere caching placement policy, and thus, Content which is in high demand is able to be resolved in a quicker fashion. This holds true, both for the Equal and Diff variants. Moreover, the number of Interests travelling through the network (Figure 5.8(a)) is maintained relatively constant across both strategies and variants, which further enforces the notion that for the same amount of Interests, better pathing is achieved by using a context-based concept, which ultimately results is lower delays for retrieval of Content.

5.3.1.3 Network Overhead

The graphs presented previously show that including context analysis in the forwarding decision making can provide better results when it comes to Interest satisfaction. However, that notion can be misleading in case the number of Interests travelling through the network is much higher than intended, as it can induce unwanted network congestion, which escalates with the number of nodes.

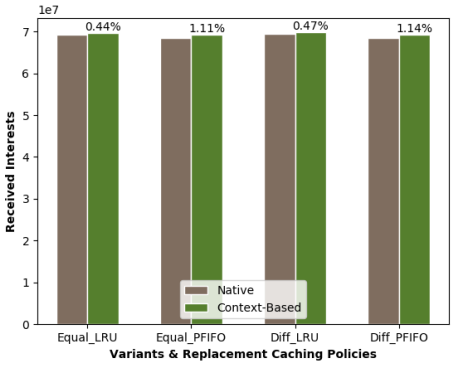


(a) Equal Cache Last Delay.

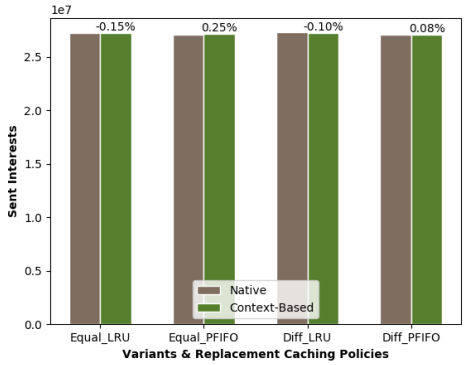


(b) Diff Cache Last Delay.

Figure 5.7: Last Delay study for both Equal and Diff variants, following a thirty minute interval observation period.



(a) In Interests.



(b) Out Interests.

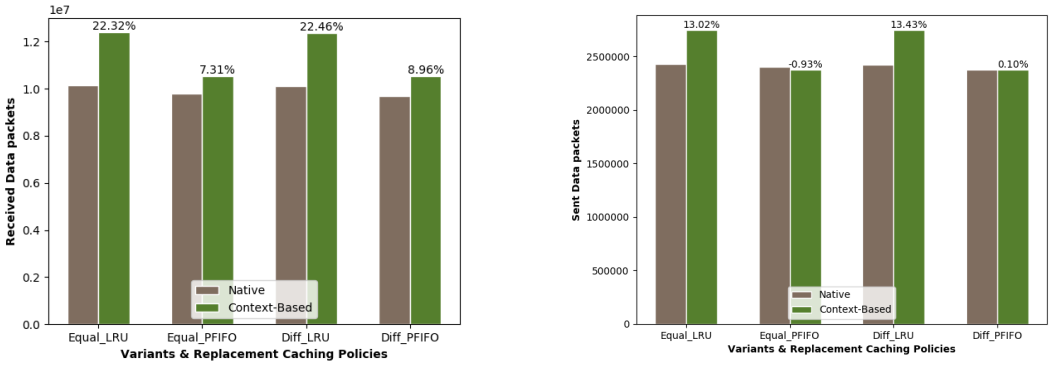
Figure 5.8: Use Case 1 - Global transmitted and received Interest packets overview.

Figures 5.8(a) and 5.8(b) show that the context-based mechanism induces a marginally higher number of Interest packets received by nodes on a global scale. This increase is expected, since new options are available for packet forwarding, allowing for a higher number of nodes to receive Interests which would not have been received by using the native strategy, which affects the registered numbers. But, as shown in Figure 5.8(a), the increment of the Interest packages was negligible.

In Figure 5.4 we noticed that, by using LRU, a higher number of cache hits could be achieved. This occurrence directly impacts the number of Interests that need to be send by nodes on a global scale, due to Content being available closer to the consumer, which explains the slightly reduced number of transmitted Interests in the context-based strategy. However, the same does not happen when using PFIFO, where the number of Interests sent to external nodes is higher. This value can be made clear by the fact that more re-transmissions will occur, thus more Interests will need to be sent to the network.

Figures 5.9(a) and 5.9(b) depict the number of Data packets travelling the network, symbolising that Interests successfully reach a Content provider. Since the number of cache hits

increases with the context-based approach, it is expected that the number of Data packets travelling the network is higher that the native strategy. In fact, this can be observed in the graphs, with exception of the PFIFO variant, in which was already noticed that less Interests were satisfied, meaning that more Interests expired before receiving the Content, which can result in dropped Data packets, thus showing in a lesser value of Data sent by nodes. The explanation for a higher count of received Data packets compared to the sent ones in the PFIFO variant, is due to the path chosen by the forwarding strategy being different than in the native one, which can be better suited for the mobility of the nodes at the time that Content is being transmitted.



(a) In Data.

(b) Out Data.

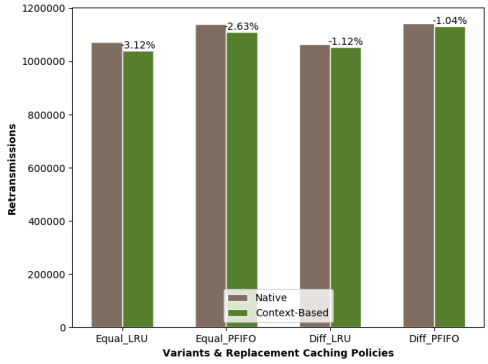
Figure 5.9: Use Case 1 - Global sent and received Data packets overview.

Figure 5.10(a) illustrates the number of packet re-transmissions enforced by consumers in an effort to receive a certain Content. Since it was already discussed that Interest satisfaction rate and cache hits increased with the use of a context-based approach, it is expected that a lower number of Interest re-transmission will occur, a fact which can be observed in all strategies compared in the mentioned graph.

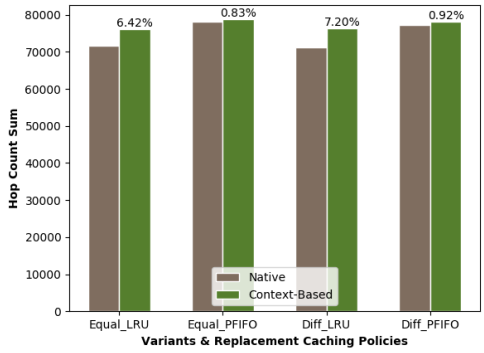
In a similar tone, due to the behaviour of the proposed forwarding strategy, by inspecting Figure 5.10(b) a conclusion can be drawn that the number of hop counts needed to resolve a certain Content increases, a fact that makes it possible for a higher number of Interests to be satisfied for the Use Case under analysis.

A further study concerning the changes to the number of hop counts needed for a Content to be resolved can be performed, by analyzing its evolution over the four hours period, in intervals of thirty minutes. This study was performed both for the native and context-aware strategies, as well as for the variants and replacement caching policies considered. However, due to presenting similar behaviour across both variants, only one on of each was considered for visibility, and can be observed in Tables 5.4 and 5.5.

A general pattern can be induced for both strategies: (i) the native approach achieves a higher amount of 1 hop communications over the course of the simulation while maintaining minimal growth in communications that require a higher number of hops; (ii) the context-aware approach achieves a growth over the course of the simulation in communications which require more than 2 hops, present less overall use of 1 hop communications than the native strategy, but allowing for the satisfaction of a higher number of Interests, specially in the



(a) Interest Re-transmissions.



(b) Interest Hop Count Sum.

Figure 5.10: Use Case 1 - Consumer related metrics.

Hours	0.5	1	1.5	2	2.5	3	3.5	4
Native 1 Hop Count	94.5	95.6	97.1	96.4	96.0	97.5	96.2	95.0
Native 2 Hops Counts	5.1	4.2	2.6	3.3	3.7	2.3	3.5	4.5
Native 3 Hops Count	0.4	0.2	0.3	0.3	0.3	0.2	0.3	0.5
	100 %							
Context-Aware 1 Hop Count	93.9	95.4	96.6	94.5	92.1	91.5	88.1	85.6
Context-Aware 2 Hops Count	5.6	4.3	3.3	4.5	6.4	6.0	8.0	10.2
Context-Aware 3 Hops Count	0.5	0.3	0.1	1.0	1.5	2.5	3.9	4.2
	100 %							

Table 5.4: Use Case 1 - Equal Cache LRU Hop Count percentage study over time.

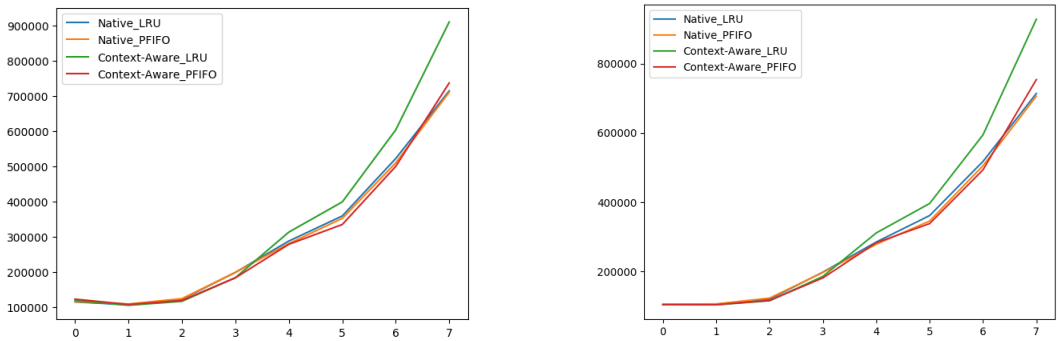
later stages, which can be observed in Figures 5.11(a) and 5.11(b).

5.3.2 Use Case 2 – Mixed requests

Use case 1 focused solely on the analysis of the performance of context-aware forwarding in regards to the dissemination of urgent Content. In contrast, Use Case 2 broadens this perspective by including Contents of lower forwarding effort, non-urgent, attempting to mimic a real network environment, which is more chaotic in nature. The fact that there is more Content available to consumers but the cache size remains constant across both Use Cases, means that more replacements will occur. As referred in the previous Use Case, for visibility, both information domains were aggregated into a single one in table 5.6. Likewise, consumer nodes will request all available Contents regarding each domain, both urgent and non-urgent, at a frequency of one Interest per second. On the other hand, producer nodes will provide five files for each Content with increasing freshness values, due to the consideration that more impactful domains will produce Content which is less relevant over a long period of time.

Hours	0.5	1	1.5	2	2.5	3	3.5	4
Native 1 Hop Count	94.6	95.9	97.1	97.0	97.1	98.6	98.4	98.6
Native 2 Hops Counts	5.3	3.9	2.8	2.8	2.7	1.3	1.5	1.3
Native 3 Hops Count	0.1	0.2	0.1	0.2	0.2	0.1	0.1	0.1
	100 %							
Context-Aware 1 Hop Count	93.9	95.7	96.1	95.9	96.3	96.6	95.5	95.3
Context-Aware 2 Hops Count	5.7	4.2	3.7	3.7	3.5	3.1	3.7	4.0
Context-Aware 3 Hops Count	0.4	0.1	0.2	0.4	0.2	0.3	0.8	0.7
	100 %							

Table 5.5: Use Case 1 - Equal Cache PFIFO Hop Count percentage study over time.



(a) Equal.

(b) Diff.

Figure 5.11: Use Case 1 - Interest Satisfaction over an interval of 30 minutes, for both variants

5.3.2.1 Satisfied Interests Rate

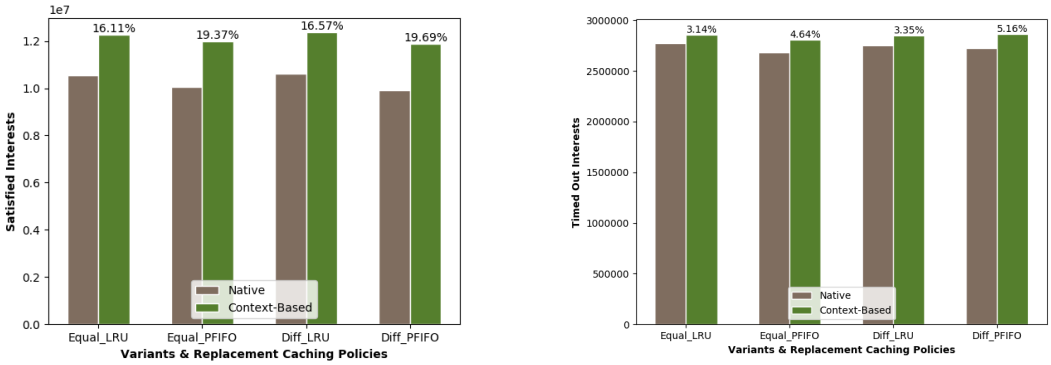
Figure 5.12(a) presents the number of Interests that successfully were satisfied, across all nodes in the network. The graph shows that there was a boost in performance when using context-awareness for both provider cache Content placement variants as well as for the two replacement policies considered. While in Use Case 1 we could observe a struggle to satisfy Interests when using PFIFO, with the introduction of non-urgent Content to the mix, the same does not occur. This allows us to reach two conclusions: (i) while the number of Contents available is higher, by having Content which is deemed more important to maintain than others, Content which in Use Case 1 would have been replaced is now allowed to remain closer to the consumer for a longer period of time; (ii) links which would not have been considered in the first Use Case, as the forwarding strategy makes use of urgency values to reach a decision, are now considered, which allow for better reception of Content.

Figure 5.12(b) illustrates that there is a higher amount of Interests timing out by using context-awareness for this Use Case. This is to be expected, due to the nature of the Use Case in itself, as referred previously, which promotes a higher rate of Content replacement, and the manner how forwarding is performed for certain Content. Since some Interests will need higher hop counts in order to discover the Data requested, specially in the first stages of simulation, there is a greater probability that expiry will occur as well as the in-network caching feature not being as efficient due to the sheer amount of Content being disseminated,

Content	Consumer (Zipf Distribution)			Producers	
	Requests	Urgency	Rate (int/sec)	Available Content	Freshness (seconds)
Emergency	All	Urgent	1.0	5	900
		Non-urgent			
Decision		Urgent			1800
		Non-urgent			
Feedback		Urgent			3000
		Non-urgent			
Interaction		Urgent			
		Non-urgent			
Information		Urgent		10	3600
		Non-urgent			

Table 5.6: Scenario 2 considered parameters.

meaning that this Content might be replaced before re-transmissions can happen.



(a) Satisfied Interests.

(b) Timed Out Interests.

Figure 5.12: Use Case 2 - Global Interest overview

As in Use Case 1, context-based forwarding allows for larger numbers of cache hits across the network, as noticeable in Figure 5.13, which is further confirmed by the increased Interest satisfaction rate.

When it comes to consumer Satisfied Interests rate, however, the same behaviour noticed before occurs in this Use Case. Figure 5.14(a) shows that the context-based implementation falls short of the native one, by satisfying less Interests over the course of the simulation. In this case, due to the nature of the Use Case in question, this circumstance is more noticeable, as there are more Contents to be requested, which promotes a higher degree of replacements in intermediate nodes. Mobility also is a factor of discussion here, due to the way Data travels to reach the consumer (in-reverse path). Since the number of satisfied Interests is reduced overall, it is expected that more timeouts will take effect, which can be perceived in Figure 5.14(b).

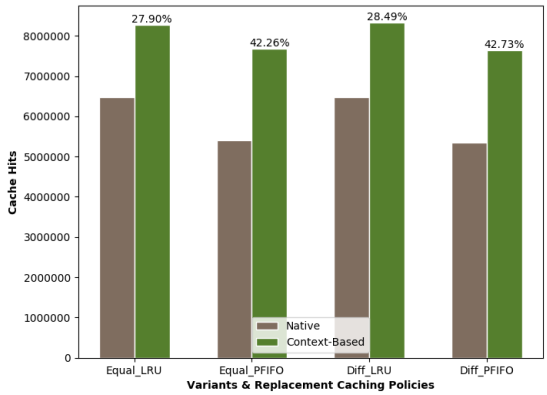
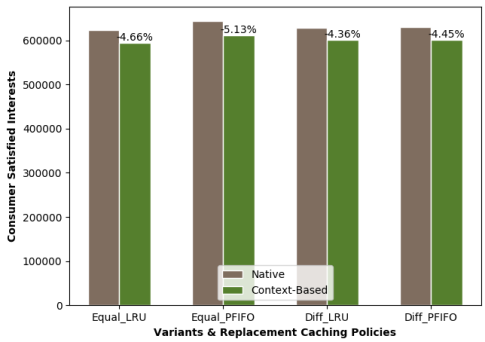
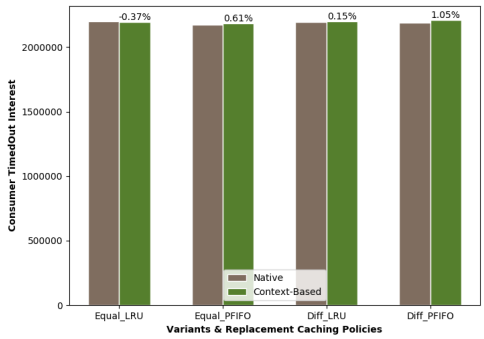


Figure 5.13: Use Case 2 - Global cache Hits overview.



(a) Satisfied Interests.



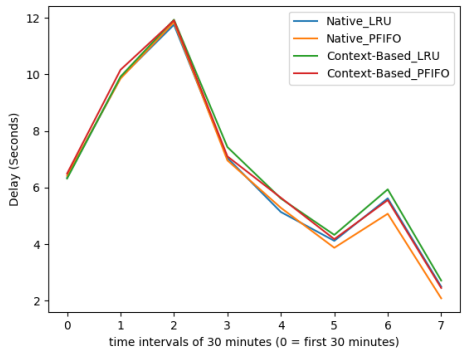
(b) Timed Out Interests.

Figure 5.14: Use Case 2 - Consumer specific Interest overview.

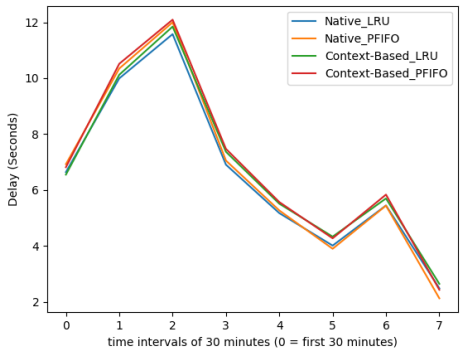
5.3.2.2 Transmission Delay

Figures 5.15(a) and 5.15(b) illustrate the average full delay needed for a first Interest to trigger a successful Data reception, for both caching replacement policies considered. Results obtained show that in this particular Use Case, which is more tumultuous in nature, the context-aware implementation induces a longer delay overall, mainly due to the number of hop counts needed to satisfy Interests possessing a higher value than the native implementation.

In contrast, the average last delay values registered in Figures 5.16(a) and 5.16(b), which do not include re-transmissions, show that the native strategy falls short of a context-based approach, a fact visible for both PFIFO and LRU replacement policies, over the course of the full simulation. On this subject, likewise in the first Use Case, a conclusion can be drawn that as more Interests successfully discover a Content provider as seen in Figures 5.12(a) and 5.13, they are stored closer to the consumer and Content in high demand is resolved in a quicker fashion.

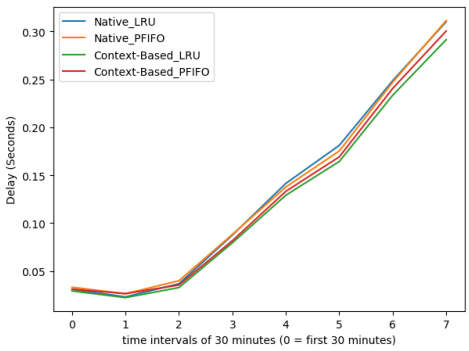


(a) Equal Cache Full Delay.

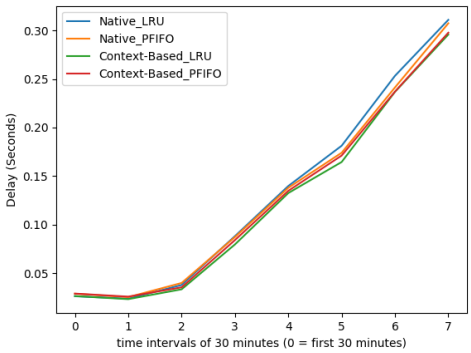


(b) Diff Cache Full Delay.

Figure 5.15: Use Case 2 - Average Full Delay study for both Equal and Diff variants, following a thirty minute interval observation period.



(a) Equal Cache Last Delay.

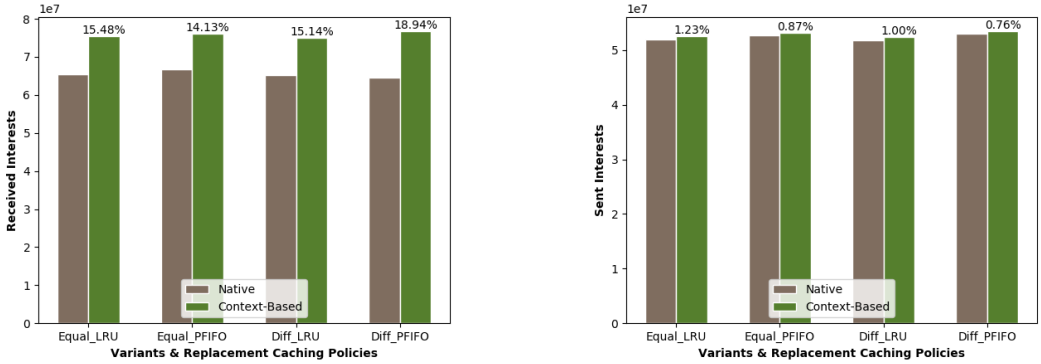


(b) Diff Cache Last Delay.

Figure 5.16: Use Case 2 - Average Last Delay study for both Equal and Diff variants, following a thirty minute interval observation period.

5.3.2.3 Network Overhead

The Figures presented before show that including context analysis in the forwarding decision making can provide better results concerning Interest satisfaction. However, as a result of the increased amount of Content travelling the network, a higher value of timed out Interests is noticeable, since re-transmissions can be less efficient than in the native approach, due to Content replacement. This is further apparent by observing the number of Interests arriving at nodes, Figure 5.17(a), which considerably surpasses the original design, in a slightly negative way. Moreover, as Interest packets are able to reach farther in the network, due to the way interfaces are selected and result on a higher hop count, it is expected that this value increases, likewise in the first Use Case, as more Content is reached by Interests, as a result of the increase of cache hits (Figure 5.13). Thus, despite this value being higher, it does not necessarily mean that unnecessary Interests are being sent to the network, which can be observed in Figure 5.17(b), where the ratio of Interests sent by nodes, is marginally higher, while achieving higher satisfaction rate.

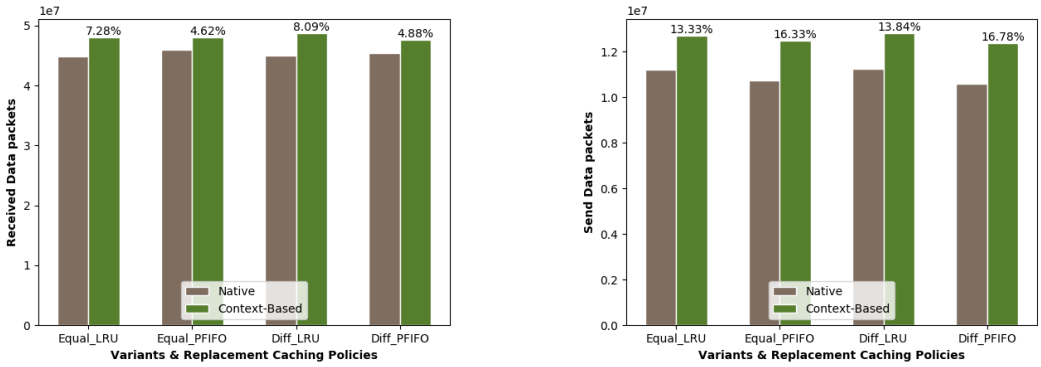


(a) In Interests. (b) Out Interests.

Figure 5.17: Use Case 2 - Global transmitted and received Interest packets overview

---

From Figure 5.13 we concluded that the number of cache hits increased significantly. This occurrence directly impacts not only the number of Interests that need to be sent by nodes on a global scale but also the Data packets that need to be sent in return. In this regard, Figures 5.18(a) and 5.18(b) show that the number of Data packets under transmission increased significantly, a notion further ascertained by the raised total of cache hits, since these two metrics work hand-in-hand. The number of Data packets sent is higher than the number of received Data packets, due to mobility itself, as the forwarding strategy makes use of in-reverse path to deliver the Content, which can not reach the consumer in case a node along that path has moved;



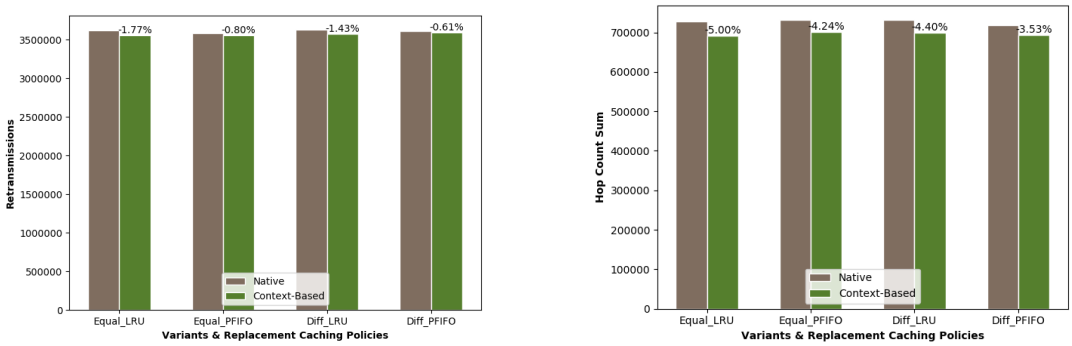
(a) In Data. (b) Out Data.

Figure 5.18: Use Case 2 - Global sent and received Data packets overview.

Figure 5.19(a) illustrates the number of packet re-transmissions enforced by consumers in an effort to receive a certain Content. Since it was already discussed that Interest satisfaction rate and cache hits increased with the use of a context-based approach, it is expected that a lower number of Interest re-transmissions will occur, a fact which can be observed in the mentioned graph.

Similarly, due to the behaviour of the proposed forwarding strategy, by inspecting Figure 5.18(b) a conclusion can be drawn that the number of hop counts needed to resolve a cer-

tain Content decreases on a global perspective, mainly due to more Content providers being reached. This results in the relay of Content closer to the consumers, which will eventually request it and be able to satisfy the Interest in a quicker fashion, specially at the later stages of the simulation, which can be observed by analysing the rate at which Interests are satisfied over, in Figure 5.20(a) and 5.20(b).



(a) Interest Re-transmissions.

(b) Interest Hop Count Sum.

Figure 5.19: Use Case 2 - Consumer related metrics.

As in the previous Use Case, an equivalent study concerning the number of hop counts needed for a Content to be resolved was performed, and while all variants were considered, due to presenting similar behaviour, only a single strategy is shown for visibility, Figures 5.7 and 5.8. It is important to keep in mind, that these hop count values only refer to Interests which successful trigger the reception of a Data packet.

Hours	0.5	1	1.5	2	2.5	3	3.5	4
Native 1 Hop Count	86.3	90.9	91.0	87.8	83.9	85.1	81.4	76.2
Native 2 Hops Count	11.3	7.7	7.0	9.6	13.4	12.6	15.7	19.2
Native 3+ Hops Count	2.4	1.4	2.0	2.6	2.7	2.3	2.9	4.6
	100%							
Context-Aware 1 Hop Count	86.2	90.1	88.5	84.7	82.4	85.1	85.8	83.2
Context-Aware 2 Hops Count	11.6	8.3	9.5	11.9	14.5	11.8	10.6	12.5
Context-Aware 3+ Hops Count	2.2	1.6	2.0	3.4	3.1	3.1	3.6	4.3
	100%							

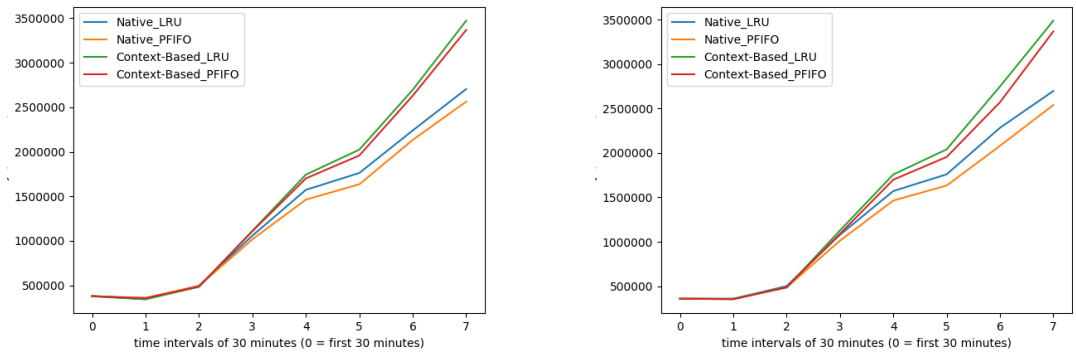
Table 5.7: Use Case 2 - Equal Cache LRU Hop Count percentage study over time

In the first hour of simulation, both strategies present similar behaviour concerning the number of hop counts needed for Interest satisfaction. Subsequently, over the following 2 hours a pattern can be induced: (i) the native approach satisfies Interests through 1 hop communications, maintaining a small growth in the number of Interests satisfied through a higher count of hops; (ii) the context-based strategy, while also satisfying most Interests through 1 hop transmissions, presents higher values of successful Interests satisfied through nodes farther away. At the later stages, the native strategy falls short of its counterpart by having to recur to more 2-hop communications to maintain satisfaction rates, while the context-based is able to satisfy a higher value through 1-hop. This behaviour was also verified

Hours	0.5	1	1.5	2	2.5	3	3.5	4
Native 1 Hop Count	86.1	91.3	90.9	88.2	84.7	86.8	84.7	84.5
Native 2 Hops Count	11.5	7.4	7.0	9.2	12.9	11.3	13.5	13.7
Native 3+ Hops Count	2.4	1.3	2.1	2.6	2.4	1.9	1.8	1.8
	100%							
Context-Aware 1 Hop Count	86.0	90.6	88.4	83.9	81.6	85.0	88.5	87.7
Context-Aware 1 Hop Count	11.6	8.1	9.6	12.7	15.0	11.8	8.8	9.0
Context-Aware 1 Hop Count	2.4	1.3	2.0	3.4	3.4	3.2	2.7	3.3
	100%							

Table 5.8: Use Case 2 - Equal Cache PFIFO hop count percentage study over time

in the first Use Case and can be observed in Figures 5.20(a) and 5.20(b), which illustrate the rate of Interests satisfied over the course of the simulation in intervals of 30 minutes. It can be concluded that, despite the node mobility in action, context-based is able to deliver more Content than the native approach.



(a) Equal. (b) Diff.

Figure 5.20: Use Case 2 - Satisfied Interests over an interval of 30 minutes, for both variants.

5.4 Final Considerations

This chapter focused on the evaluation of the context-based forwarding strategy here proposed, which main purpose was to address Content delivery in mobile NDN, through experimentation using ndnSIM software and taking advantage of Data retrieved from a real vehicular ad hoc network. To that extent, the Use Cases under evaluation were presented and described, as well as an explanation of the metrics considered important for a complete analysis of the performance of the proposed strategy.

The results obtained show that the context-based forwarding strategy improves the satisfied Interests rate while maintaining network overhead and delays at an acceptable threshold, with LRU possessing an edge over the PFIFO cache replacement policy. From both Use Cases evaluation we can infer that the setting of some parameters can be identified as important in the configuration of the proposed strategy, due to the direct impact on performance, mainly

the specification of the urgency value for each Content and the application domains, via packet naming.

Chapter 6

Conclusion and Future Work

Existing forwarding mechanisms following the ICN paradigm can achieve good results concerning issues such as redundancy, bandwidth usage and delay management that prove this innovative design is better suited for mobility aware network environments as opposed to the traditional TCP/IP architectures. While the entire framework was built on the concept of forwarding Content identified by names, there is still not much study regarding the potential of using them as an actual parameter in the forwarding decisions to be performed on a hop-by-hop basis for each request under analysis in the nodes which comprise the network.

The main goal of this dissertation was the study, design, implementation and evaluation of forwarding strategy for ICNs, more specifically, NDNs addressing mobility concerns, which takes input from both packet context and neighbourhood status to weigh in the decision making process. On one hand, it makes an effort to understand to which context a certain name belongs and infers its degree of urgency by the impact of packet loss in the application layer. On the other hand, it evaluates the neighbourhood considering which types of nodes are available per interface, in order to understand the feasibility of forwarding Content through specific link types.

A number of challenges appeared during the development of this work. Several were related to the understanding of how the NDN modules were integrated in the simulator and which adjustments needed to be made to the existing structure to allow for network topology variation over the course of simulations. Furthermore, it was needed to review the impact of these adjustments on the analysis of a new forwarding strategy, which makes use of the already established structures, in order to guarantee they were independent from the forwarding strategy plane, and thus, independent of the decision making. However, these challenges were overcome and the proposed context-based strategy was considered stable for evaluation.

After analysing the behaviour of the proposed forwarding strategy in the scenarios presented it was possible to draw several conclusions. There is a clear improvement in terms of satisfied Interests rate across the overall network when compared to the existing Least-Cost approach. In contrast, there is a decrease in the rate of satisfied Interests on the nodes originating the requests. On the subject of network delay, a slight upgrade is noticeable on the time needed for an Interest to trigger a successful Data reception, although this enhancement does not exist when re-transmissions are inserted into the equation. Furthermore, the forwarding strategy is able to maintain a stable number of packets flowing through the network, presenting higher values mostly under the form of Data packets, which further enforces the

higher rate of Interests satisfied, as more Content circulates throughout the nodes, and thus, do not contribute significantly towards network congestion naturally.

With all the results gathered, it was then possible to draw some conclusions about the proposed context-based strategy performance. All summed up, it can be assumed that while not entirely positive results were achieved with the developed implementation, a good compromise was drawn given that Content dissemination is achieved with good satisfaction rate and delay times. This fact makes it possible to consider a context-based forwarding approach as a good choice for Content distribution in mobile ICNs, which was the primary goal of this dissertation.

6.1 Future Work

The forwarding strategy proposed raised some concerns over the course of its implementation, as well as did not explore the full potential of mobility patterns and how they can impact the network performance. Thus, not all possible optional parameters were analysed and evaluated whereby several improvements and future topics of research should be addressed. Future work should be focused on the following topics:

- **Improve packet flooding flow both for Interest and Data packets**. This dissertation raises concerns towards the unavoidable packet flooding in wireless connections. A considered proposal of a method to address this is to select one specific node to forward information in the same neighbourhood, similarly to existing clustering methods, in an effort to reduce the network congestion from the current broadcast communication.

- **Move urgency metric out of application layer** In the context of this work, the urgency parameter was assumed as being of the domain of the application layer. This poses a great issue from a security perspective due to every application having the choice to establish an arbitrary value to the Content and therefore influence forwarding decisions. Thus, this concept should be moved out of the application layer by only allowing specific authenticated nodes to attribute urgency to Interests under dissemination, in an automated manner.

- **Bring mobility pattern to the equation** The current implementation takes advantage of the Content context and the neighbourhood status, but could be further improved with mobility patterns in order to induce future node conditions and perform better decisions.

- **Evolution to a real network testing and evaluation**: even though the proposed forwarding strategy was evaluated using Data from a real vehicular ad hoc network, it was tested in an emulation environment. Therefore, the strategy should be tested on a real testbed in order to assess its behaviour.

Bibliography

[1] Van Jacobson, Diana K. Smetters, James D. Thornton, Michael F. Plass, Nicholas H. Briggs, and Rebecca L. Braynard. Networking Named Content. In *Proceedings of the 5th International Conference on Emerging Networking Experiments and Technologies*, CoNEXT '09, pages 1–12, New York, NY, USA, 2009. ACM.

[2] Safdar Hussain Bouk, Syed Hassan Ahmed, and Dongkyun Kim. Hierarchical and hash based naming with Compact Trie name management scheme for Vehicular Content Centric Networks. *Computer Communications*, 71:73 – 83, 2015.

[3] Lixia Zhang, Alexander Afanasyev, Jeffrey Burke, Van Jacobson, Kc claffy, Patrick Crowley, Christos Papadopoulos, Lan Wang, and Beichuan Zhang. Named Data Networking. *SIGCOMM Comput. Commun. Rev.*, 44(3):66–73, July 2014.

[4] M. F. Bari, S. R. Chowdhury, R. Ahmed, R. Boutaba, and B. Mathieu. A survey of naming and routing in information-centric networks. *IEEE Communications Magazine*, 50(12):44–53, December 2012.

[5] M. Amadeo, C. Campolo, and A. Molinaro. Information-centric networking for connected vehicles: a survey and future perspectives. *IEEE Communications Magazine*, 54(2):98–104, February 2016.

[6] Spyridon Mastorakis, Alexander Afanasyev, and Lixia Zhang. On the Evolution of ndnSIM: an Open-Source Simulator for NDN Experimentation. *ACM Computer Communication Review*, July 2017.

[7] Spyridon Mastorakis, Alexander Afanasyev, Ilya Moiseenko, and Lixia Zhang. ndnSIM 2: An updated NDN simulator for NS-3. Technical Report NDN-0028, Revision 2, NDN, November 2016.

[8] C. Fang, H. Yao, Z. Wang, W. Wu, X. Jin, and F. R. Yu. A Survey of Mobile Information-Centric Networking: Research Issues and Challenges. *IEEE Communications Surveys Tutorials*, 20(3):2353–2371, thirdquarter 2018.

[9] J. Pan, S. Paul, and R. Jain. A survey of the research on future internet architectures. *IEEE Communications Magazine*, 49(7):26–36, July 2011.

[10] B. Ahlgren, C. Dannewitz, C. Imbrenda, D. Kutscher, and B. Ohlman. A survey of information-centric networking. *IEEE Communications Magazine*, 50(7):26–36, July 2012.

[11] Boubakr Nour, Kashif Sharif, Fan Li, Sujit Biswas, Hassine Moungla, Mohsen Guizani, and Yu Wang. A survey of Internet of Things communication using ICN: A use case perspective. *Computer Communications*, 142-143:95 – 123, 2019.

[12] Elisa Mannes and Carlos Maziero. Naming Content on the Network Layer: A Security Analysis of the Information-Centric Network Model. *ACM Comput. Surv.*, 52(3), June 2019.

[13] Stephen Farrell et. al. Naming Things with Hashes, RFC 6920. Technical report, 2013.

[14] Sobia Arshad, Muhammad Awais Azam, Mubashir Husain Rehmani, and Jonathan Loo. Recent Advances in Information-Centric Networking-Based Internet of Things (ICN-IoT). *IEEE Internet of Things Journal*, 6(2):2128–2158, April 2019.

[15] S. Arshad, B. Shahzaad, M. A. Azam, J. Loo, S. H. Ahmed, and S. Aslam. Hierarchical and Flat-Based Hybrid Naming Scheme in Content-Centric Networks of Things. *IEEE Internet of Things Journal*, 5(2):1070–1080, April 2018.

[16] S. S. Adhatarao, J. Chen, M. Arumaithurai, X. Fu, and K. K. Ramakrishnan. Comparison of naming schema in ICN. In *2016 IEEE International Symposium on Local and Metropolitan Area Networks (LANMAN)*, pages 1–6, June 2016.

[17] I. U. Din, S. Hassan, M. K. Khan, M. Guizani, O. Ghazali, and A. Habbal. Caching in Information-Centric Networking: Strategies, Challenges, and Future Research Directions. *IEEE Communications Surveys Tutorials*, 20(2):1443–1474, Secondquarter 2018.

[18] Ioannis Psaras, Wei Koong Chai, and George Pavlou. Probabilistic In-network Caching for Information-centric Networks. In *Proceedings of the Second Edition of the ICN Workshop on Information-centric Networking*, ICN '12, pages 55–60, New York, NY, USA, 2012. ACM.

[19] Cheng Yi, Alexander Afanasyev, Ilya Moiseenko, Lan Wang, Beichuan Zhang, and Lixia Zhang. A case for stateful forwarding plane. *Computer Communications*, 36(7):779 – 791, 2013.

[20] Cheng Yi, Alexander Afanasyev, Lan Wang, Beichuan Zhang, and Lixia Zhang. Adaptive Forwarding in Named Data Networking. *SIGCOMM Comput. Commun. Rev.*, 42(3):62–67, June 2012.

[21] Teemu Koponen, Mohit Chawla, Byung-Gon Chun, Andrey Ermolinskiy, Kye Hyun Kim, Scott Shenker, and Ion Stoica. A Data-oriented (and Beyond) Network Architecture. *SIGCOMM Comput. Commun. Rev.*, 37(4):181–192, August 2007.

[22] Christian Dannewitz, Dirk Kutscher, Börje Ohlman, Stephen Farrell, Bengt Ahlgren, and Holger Karl. Network of Information (NetInf) - An information-centric networking architecture. *Computer Communications*, 36(7):721 – 735, 2013.

[23] Nikos Fotiou, Pekka Nikander, Dirk Trossen, and George C. Polyzos. Developing Information Networking Further: From PSIRP to PURSUIT. In Ioannis Tomkos, Christos J. Bouras, Georgios Ellinas, Panagiotis Demestichas, and Prasun Sinha, editors, *Broadband Communications, Networks, and Systems*, pages 1–13, Berlin, Heidelberg, 2012. Springer Berlin Heidelberg.

[24] Dipankar Raychaudhuri, Kiran Nagaraja, and Arun Venkataramani. MobilityFirst: A Robust and Trustworthy Mobility-centric Architecture for the Future Internet. *SIGMOBILE Mob. Comput. Commun. Rev.*, 16(3):2–13, December 2012.

[25] L. Nassar, A. Jundi, K. Golestan, F. Sattar, F. Karray, M. Kamel, and S. Boumaiza. Vehicular Ad-hoc Networks(VANETs): Capabilities, Challenges in Context-Aware Processing and Communication Gateway. In Mohamed Kamel, Fakhri Karray, and Hani Hagras, editors, *Autonomous and Intelligent Systems*, pages 42–49, Berlin, Heidelberg, 2012. Springer Berlin Heidelberg.

[26] Saif Al-Sultan, Moath M. Al-Doori, Ali H. Al-Bayatti, and Hussien Zedan. A Comprehensive Survey on Vehicular Ad Hoc Network. *J. Netw. Comput. Appl.*, 37:380–392, January 2014.

[27] F. Dressler, H. Hartenstein, O. Altintas, and O. K. Tonguz. Inter-vehicle communication: Quo vadis. *IEEE Communications Magazine*, 52(6):170–177, June 2014.

[28] Mohamed Nidhal Mejri, Jalel Ben-Othman, and Mohamed Hamdi. Survey on VANET security challenges and possible cryptographic solutions. *Vehicular Communications*, 1(2):53 – 66, April 2014.

[29] 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)*, pages 1–6, Sep. 2016.

[30] C. Fang, H. Yao, Z. Wang, W. Wu, X. Jin, and F. R. Yu. A Survey of Mobile Information-Centric Networking: Research Issues and Challenges. *IEEE Communications Surveys Tutorials*, 20(3):2353–2371, thirdquarter 2018.

[31] L. Han, S. Kang, H. Kim, and H. P. In. Adaptive Retransmission Scheme for Video Streaming over Content-Centric Wireless Networks. *IEEE Communications Letters*, 17(6):1292–1295, June 2013.

[32] M. Amadeo, C. Campolo, and A. Molinaro. Design and analysis of a transport-level solution for content-centric VANETs. In *2013 IEEE International Conference on Communications Workshops (ICC)*, pages 532–537, June 2013.

[33] G. Xylomenos, X. Vasilakos, C. Tsilopoulos, V. A. Siris, and G. C. Polyzos. Caching and mobility support in a publish-subscribe internet architecture. *IEEE Communications Magazine*, 50(7):52–58, July 2012.

[34] Do-hyung Kim, Jong-hwan Kim, Yu-sung Kim, Hyun-soo Yoon, and Ikjun Yeom. Mobility Support in Content Centric Networks. In *Proceedings of the Second Edition of the ICN Workshop on Information-centric Networking*, ICN '12, pages 13–18, New York, NY, USA, 2012. ACM.

[35] Do-hyung Kim, Jong-hwan Kim, Yu-sung Kim, Hyun-soo Yoon, and Ikjun Yeom. End-to-end Mobility Support in Content Centric Networks. *Int. J. Commun. Syst.*, 28(6):1151–1167, April 2015.

[36] Divya Saxena, Vaskar Raychoudhury, Neeraj Suri, Christian Becker, and Jiannong Cao. Named Data Networking: A survey. *Computer Science Review*, 19:15 – 55, 2016.

[37] Yaogong Wang, Natalya Rozhnova, Ashok Narayanan, David Oran, and Injong Rhee. An Improved Hop-by-hop Interest Shaper for Congestion Control in Named Data Networking. *SIGCOMM Comput. Commun. Rev.*, 43(4):55–60, August 2013.

[38] Xiaoke Jiang and Jun Bi. Interest Set Mechanism to Improve the Transport of Named Data Networking. volume 43, pages 515–516, 08 2013.

[39] Cheng Yi, Alexander Afanasyev, Lan Wang, Beichuan Zhang, and Lixia Zhang. Adaptive Forwarding in Named Data Networking. *SIGCOMM Comput. Commun. Rev.*, 42(3):62–67, June 2012.

[40] X. Zeng and Z.H. Gao. Rank-based routing strategy for Named Data Network. *Applied Mechanics and Materials*, 543-547:3320–3323, 2014. cited By 3.

[41] Cheng Yi, Alexander Afanasyev, Ilya Moiseenko, Lan Wang, Beichuan Zhang, and Lixia Zhang. A case for stateful forwarding plane. *Computer Communications*, 36(7):779 – 791, 2013.

[42] Lucas Wang, Alexander Afanasyev, Romain Kuntz, Rama Vuyyuru, Ryuji Wakikawa, and Lixia Zhang. Rapid Traffic Information Dissemination Using Named Data. In *Proceedings of the 1st ACM Workshop on Emerging Name-Oriented Mobile Networking Design - Architecture, Algorithms, and Applications*, NoM '12, pages 7–12, New York, NY, USA, 2012. ACM.

[43] Marica Amadeo, Claudia Campolo, and Antonella Molinaro. Enhancing content-centric networking for vehicular environments. *Computer Networks*, 57(16):3222 – 3234, 2013. Information Centric Networking.

[44] You Lu, Biao Zhou, Lung-Chih Tung, Mario Gerla, Ashwin Ramesh, and Lohith Nagaraja. Energy-efficient Content Retrieval in Mobile Cloud. In *Proceedings of the Second ACM SIGCOMM Workshop on Mobile Cloud Computing*, MCC '13, pages 21–26, New York, NY, USA, 2013. ACM.

[45] Y. Yu, R. B. Dilmaghani, S. Calo, M. Y. Sanadidi, and M. Gerla. Interest propagation in named data manets. In *2013 International Conference on Computing, Networking and Communications (ICNC)*, pages 1118–1122, Jan 2013.

[46] Marica Amadeo, Antonella Molinaro, and Giuseppe Ruggeri. E-CHANET: Routing, forwarding and transport in Information-Centric multihop wireless networks. *Computer Communications*, 36(7):792 – 803, 2013.

[47] J. J. Garcia-Luna-Aceves and M. Mirzazad-Barijough. A light-weight forwarding plane for content-centric networks. In *2016 International Conference on Computing, Networking and Communications (ICNC)*, pages 1–7, Feb 2016.

[48] G. Zhang, H. Li, T. Zhang, D. Li, and L. Xu. A multi-path forwarding strategy for content-centric networking. In *2015 IEEE/CIC International Conference on Communications in China (ICCC)*, pages 1–6, Nov 2015.

[49] Paulo Duarte, Joaquim Macedo, Antonio Duarte Costa, Maria João Nicolau, and Alexandre Santos. A Probabilistic Interest Forwarding Protocol for Named Data Delay Tolerant

Networks. In Nathalie Mitton, Melike Erol Kantarci, Antoine Gallais, and Symeon Papavassiliou, editors, *Ad Hoc Networks*, pages 94–107, Cham, 2015. Springer International Publishing.

[50] Christos Tsilopoulos and George Xylomenos. Adaptive semi-stateless forwarding for Content-Centric Networks. *Computer Networks*, 153:73 – 85, 2019.

[51] C. Ghasemi, H. Yousefi, K. G. Shin, and B. Zhang. MUCA: New Routing for Named Data Networking. In *2018 IFIP Networking Conference (IFIP Networking) and Workshops*, pages 289–297, May 2018.

[52] Gareth Tyson, Nishanth Sastry, Ivica Rimac, Ruben Cuevas, and Andreas 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, pages 1–6, New York, NY, USA, 2012. ACM.

[53] A. Compagno, M. Conti, C. Ghali, and G. Tsudik. To NACK or Not to NACK? Negative Acknowledgments in Information-Centric Networking. In *2015 24th International Conference on Computer Communication and Networks (ICCCN)*, pages 1–10, Aug 2015.

[54] L. Wang, V. Lehman, A. K. M. Mahmudul Hoque, B. Zhang, Y. Yu, and L. Zhang. A Secure Link State Routing Protocol for NDN. *IEEE Access*, 6:10470–10482, 2018.

[55] Marica Amadeo, Claudia Campolo, and Antonella Molinaro. Forwarding strategies in named data wireless ad hoc networks: Design and evaluation. *Journal of Network and Computer Applications*, 50:148 – 158, 2015.

[56] Alexander Afanasyev, Spyridon Mastorakis, Ilya Moiseenko, and Lixia Zhang. ndnSim, 2019.

[57] P. M. Santos, J. G. P. Rodrigues, S. B. Cruz, T. Lourenço, P. M. d'Orey, Y. Luis, C. Rocha, S. Sousa, S. Crisóstomo, C. Queirós, S. Sargento, A. Aguiar, and J. Barros. PortoLivingLab: an IoT-based Sensing Platform for Smart Cities. *IEEE Internet of Things Journal*, PP(99):1–1, 2018.

[58] A. Ioannou and S. Weber. A Survey of Caching Policies and Forwarding Mechanisms in Information-Centric Networking. *IEEE Communications Surveys Tutorials*, 18(4):2847–2886, Fourthquarter 2016.