



Article

Insights from the Experimentation of Named Data Networks in Mobile Wireless Environments

Luís Gameiro ¹, Carlos Senna ¹ and Miguel Luís ^{1,2,*}

¹ Instituto de Telecomunicações, 3810-193 Aveiro, Portugal; luismiguelgameiro@ua.pt (L.G.); cr.senna@av.it.pt (C.S.)

² ISEL—Instituto Superior de Engenharia de Lisboa, Instituto Politécnico de Lisboa, 1959-007 Lisboa, Portugal

* Correspondence: nmal@av.it.pt

Abstract: The Information-Centric Network (ICN) paradigm has been touted as one of the candidates for the Internet of the future, where the Named Data Network (NDN) architecture is the one leading the way. Despite the large amount of works published in the literature targeting new implementations of such architecture, covering different network topologies and use cases, there are few NDN implementations in real networks. Moreover, most of these real-world NDN implementations, especially those addressing wireless and wired communication channels, are at a small scale, in laboratory environments. In this work, we evaluate the performance of an NDN-based implementation in a mobile wireless network, as part of a smart city infrastructure, making use of multiple wireless interfaces. We start by showing how we have implemented the NDN stack in current network nodes of the smart city infrastructure, following a hybrid solution where both TCP/IP and NDN paradigms can coexist. The implementation is evaluated in three scenarios, targeting different situations: mobility, the simultaneous use of different wireless interfaces and the network characteristics. The results show that our implementation works properly and insights about the correct NDN parameterization are derived.



Citation: Gameiro, L.; Senna, C.; Luís, M. Insights from the Experimentation of Named Data Networks in Mobile Wireless Environments. *Future Internet* **2022**, *14*, 196. <https://doi.org/10.3390/fi14070196>

Academic Editors: José Carlos Lopez-Ardao, Miguel Rodríguez Pérez and Sergio Herrería Alonso

Received: 8 June 2022

Accepted: 24 June 2022

Published: 27 June 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

Keywords: mobile networks; Named Data Networks; performance evaluation; smart city

1. Introduction

Information Centric Networking (ICN) has been considered an alternative to the current Internet architecture, based on the TCP/IP stack, to overcome the drawbacks of host-centric architectures when applied to mobile Internet-of-Things (IoT) networks [1]. Such drawbacks are the frequent topology changes and intermittent connectivity caused by the mobility of the network nodes. 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. Therefore, ICN enables the deployment of in-network caching and content replication thus facilitating the efficient and timely delivery of information. Additionally, the publish-subscribe concept adopted by the ICN architecture, allied to a location-independent data identification, provides an efficient support for users' mobility.

Amongst the several ICN architectures, the one receiving more attention by the research community is the Named Data Network (NDN) [2], a pull-based network system where the Consumer requests the Content by sending an Interest packet. This packet is relayed by the network nodes until it reaches the Content, either at the Producer or cached at some node on the network. It is up to the node that owns the Content to send it along the reverse path, taken by the Interest, until it reaches the interested Consumer. Such architecture has been designed for infrastructured network topologies, with point-to-point links, and its application in mobile wireless environments requires the modification of some of the heuristics used in the NDN, especially those related with the Content forwarding [3]. Similarly, the use of the NDN architecture in a publish-subscribe communication paradigm is not straightforward because each data packet must precede an Interest packet [4].

Despite the touted virtues of the NDN paradigm, it is rare to find implementations or testbeds where such architecture can be tested under the same conditions as TCP/IP-based networks. The majority of the evaluation and comparison works based on non-simulation scenarios consider solely wired network infrastructures, or in a small scale. To the best of our knowledge there are no ICN/NDN implementations at city-scale with both wired and wireless communication links.

To address this gap, we present a city-scale implementation of an NDN architecture, optimized to support both single-data-request and publish-subscribe communication paradigms, and the support for multiple radio access technologies in a mobile environment. Such implementation is currently installed in the communications infrastructure of the Aveiro Tech City Living Lab (ATCLL) (<https://aveiro-living-lab.it.pt/> accessed on 7 June 2022), and is open for experimentation, development, test and/or demonstration of concepts, products or services targeting the Information and Communications Technology (ICT) topics. In this work, we explain the steps needed to implement an NDN architecture in such type of environments, where network nodes share multiple radio access technologies, such as ITS-G5, IEEE 802.11, LoRa and cellular 4G/5G, and we discuss its performance in real-world conditions, targeting the dissemination of a given Content to mobile users.

In summary, the main contributions of this work are:

- The deployment of an NDN architecture over a real city-scale wireless communication infrastructure;
- The performance assessment of the NDN architecture in the presence of multiple radio access communication technologies;
- The analysis of how several network characteristics impact the performance of the NDN architecture in wireless mobile environments.

The remainder of this paper is organized as follows. Section 2 overviews the related work about deployment of ICN-based solutions in real hardware devices, and consequently, in non-simulated environments. Section 3 presents the main features of our NDN-based solution and describes how we have deployed it on the ATCLL network. Section 4 presents and discusses the performance results of our solution targeting the dissemination of contents in a mobile network of the ATCLL. Finally, Section 5 concludes the paper and discusses future work.

2. Related Work

In the literature one can find several works on the deployment of solutions based on the ICN architecture, but a small number addresses large-scale infrastructures [5]. In this section, we overview the most relevant related works focused on the experimentation of ICN and NDN architectures in testbeds.

Jacobson et al. [2,6,7] introduced in 2009 the Content-Centered Network (CCN), a communication architecture built on named data, and compared it with an IP-based networks. In their comparison, several metrics were considered such as data transfer efficiency, content delivery efficiency, and Voice-over-CCN. Still, the scenario in use was a small controlled testbed in a laboratory environment.

In [8], the authors discuss the comparison between the content Cache-supported NDN architecture cloudlets with location-oriented TCP/IP architecture. The work considers the placement of cloudlets at the edge of an NDN network, close to end users. They used a virtual tour guide that provides audio and visual content to tourists in popular locations as an application scenario. However, the testbed, consisting of a standard Dell Ubuntu 16.04 laptop to host the NDN cloudlet server and five Android phones as clients, seems very simple being far from representing a city's communications infrastructure, without considering any multi-radio access technology.

In [9], the authors have repeated the experiments in [6] on a more complex communications infrastructure. They interconnect through the Internet some Consumers and a router to a Producer that is placed geographically far. They deploy the CCN over UDP to establish the

connection between the two environments (CCN and Internet). The results are similar to those obtained by [6].

The solution proposed in [10], the Named-data Link State Routing protocol (NLSR), is a protocol for intra-domain routing in NDN. The NLSR, an application level protocol similar to many IP routing protocols, uses NDN's Interest/Data packets to disseminate routing updates over network. The authors made an evaluation in terms of CPU processing time, routing convergence time, and forwarding plane performance using Mini-NDN, an emulation tool based on Mininet.

In [11] the authors propose and evaluate a scalable framework for lightweight authentication and hierarchical routing in NDN IoT. They demonstrate the scalability of their solution in a simulated dense environment with up to 40,000 nodes/km² whose results show an average onboarding convergence time of around 250 s and overhead of less than 20 kibibytes per node. However, they do not test their solution in a real environment.

In [12], the authors present NDN as a solution for in-vehicle communication. The solution was deployed on an emulated automotive testbed of Raspberry Pis with Controller Area Network (CAN) interfaces and tests were performed using network traces datasets. In the same direction, the work in [13] discussed a comparative simulation-based study of these NDN deployments over Wi-Fi 6 for Internet-of-Vehicles using real vehicular traces. Both papers do not present a real NDN deployment.

The work in [14] presents an NDN solution over ZigBee that evaluated, through three different scenarios, the suitability and ease of use of NDN in the IoT context. They compared their NDN implementation over ZigBee to the basic wireless NDN over Wi-Fi communication that uses the UDP/IP protocol stack to send NDN packets. In their tests, they tried different sets of Interest exchange rates and data packet sizes. They have measured the total time spent by each exchange (from sending the Interest to receiving the data) and also the ratio of Interests satisfied in each pool. Such tests were performed in controlled environments far from what is expected in a real smart city infrastructure.

There are some solutions for the well-known multiplication of interest and data packets in NDN networks over Wi-Fi. In [15], the authors proposed an Access Point-based proxy of Interest mechanism to mitigate WLAN channel competition. In addition, they have proposed a layer-based NDN WLAN multicast data rate selection mechanism for adaptive video streaming to improve the video bit rate. It is an interesting solution but it was only tested through simulation, such as proposed by [16]. In [17,18] follow the same line of validation. In [18], the authors use a small physical testbed consisting of an Access Point and 20 Raspberry Pi-based Wi-Fi clients. Ghasemi et al. [19] present a solution for adaptive video streaming service over NDN as a public ICN/NDN service for daily Internet users. They also show their deployment on the global NDN testbed. The authors discuss results from their implementation solution that presents a reasonable Quality-of-Experience of the service over the global NDN testbed using the videos of the NDN website. The paper [20] proposes an architecture based on NDN and discusses its main functionalities for ITS in smart cities. Still, this paper is purely discussion without any implementation.

Our testbed offers a set of different alternatives in relation to the solutions discussed before. We implemented NDN in a city-scale network that operates simultaneously with the TCP/IP stack without any interference between them. Our container-based solution is scalable and allows the inclusion of other network paradigms transparently, on-the-fly and without any interruption in the networks in operation. Furthermore, to the best of our knowledge, there are no NDN implementations with multiple and heterogeneous radio access interfaces per network node of this scale.

3. Integrating NDNs in Smart Cities

Our goal was to provide a city-scale NDN-based infrastructure in the city of Aveiro, Portugal. Aveiro already has an Internet-based communication infrastructure distributed throughout the city as part of the Aveiro Tech City Living Lab (ATCLL). This metropolitan network provides various communication technologies, such as ITS-G5, Cellular (4G/5G),

IEEE 802.11 (Wi-Fi) and LoRa, which are used to offer various services to several verticals. One of these verticals is the vehicular domain, where vehicles can communicate with each other and with the infrastructure for safety and/or entertainment purposes [21]. Thus, the nodes of the vehicular network, On-Board Units (OBUs) and Road Side Units (RSUs), were modified to include the NDN stack without harming the current Internet stack in operation (TCP/IP), allowing the operation of both paradigms at the same time.

In the next sections we briefly describe the characteristics of Named Data Networks and their differences to the TCP/IP-based architectures. Then we quickly present the communication infrastructure installed in the ATCLL, and finally we detail how we have deployed our NDN solution in the ATCLL network equipments, highlighting the peaceful coexistence between the two paradigms.

3.1. Named Data Networks

Currently, the Internet communication model TCP/IP is based on a host-centric network vision paradigm, where the request for a given content begins by locating and identifying the node in which the content is stored, followed by the establishment of a connection from which the communication of the request to acquire the content will be made, which is repeated for each request made by a consumer, as seen in Figure 1.

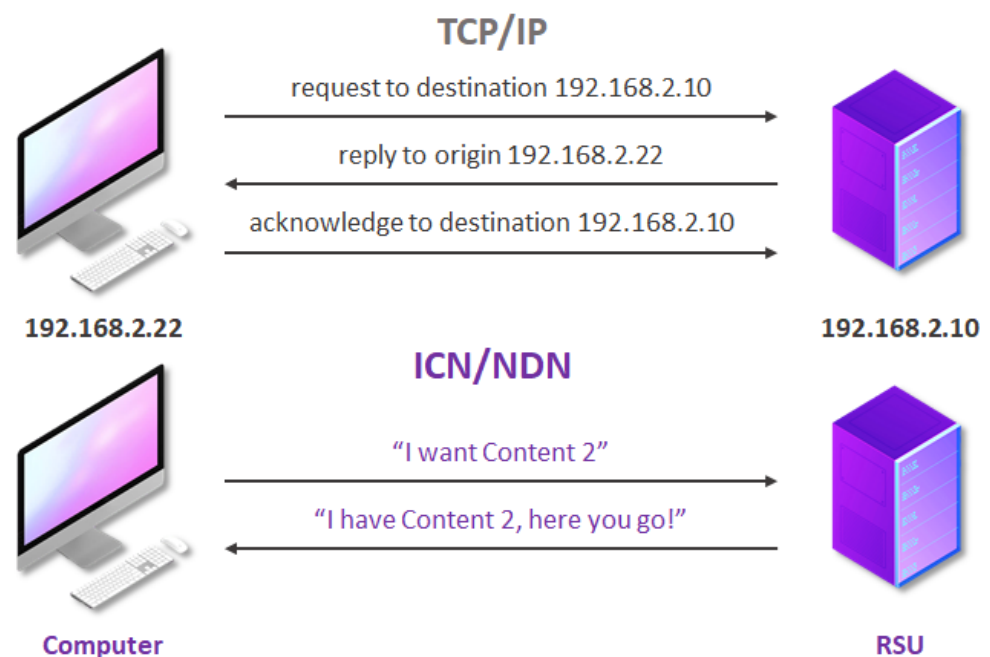


Figure 1. TCP/IP vs. NDN paradigm.

This type of model was not developed taking into account some requirements that are of interest to the IoT, such as the large number of connected devices, node mobility support, in-network caching or the security of the transmitted contents, but was rather a model that only considered the connection between a limited number of devices/computers. Some approaches have been proposed by many communities to solve these problems and needs, one of them is NDN, which places information dissemination at the centre of the network layer.

In NDN architectures, the node interested in some content (Consumer) requests it from its neighbors through its unique name without any indication of location (IP for example). The network nodes disseminate this request until it reaches the node where the desired content is located, a Producer or any other node with the Content on its cache. Thus, communication between nodes follows a Request-Response exchange model with native multicast support, which is quite efficient in mobile environments [3]. The Producer

sends the requested Content using the opposite path to that taken by the Interest, following a receiver-oriented approach.

In an NDN architecture, the Consumer requests Contents without knowing the providing host, and the communication follows a receiver-driven approach, i.e., the data follow the reverse route of the request (Figure 2a). The system is then responsible for mapping the requested data and its location. An NDN architecture is characterized by the adopted approach on each of the following key functionalities: naming and name resolution, routing and forwarding, and caching [22]. NDN nodes use three structures in the Forwarding process (Figure 2b). The Forwarding Information Base (FIB) contains the names of a given Content, provided by the routing protocol, and associates them with the interfaces on which they were received. In addition to the FIB, NDN nodes use two other tables: Pending Interest Table (PIT) and Content Store (CS). Each PIT entry contains the name of the Interest packet and the respective interfaces through which it arrived at the node. On the other hand, the CS stores the data packets and works as a buffer managed by caching policies [3,23].

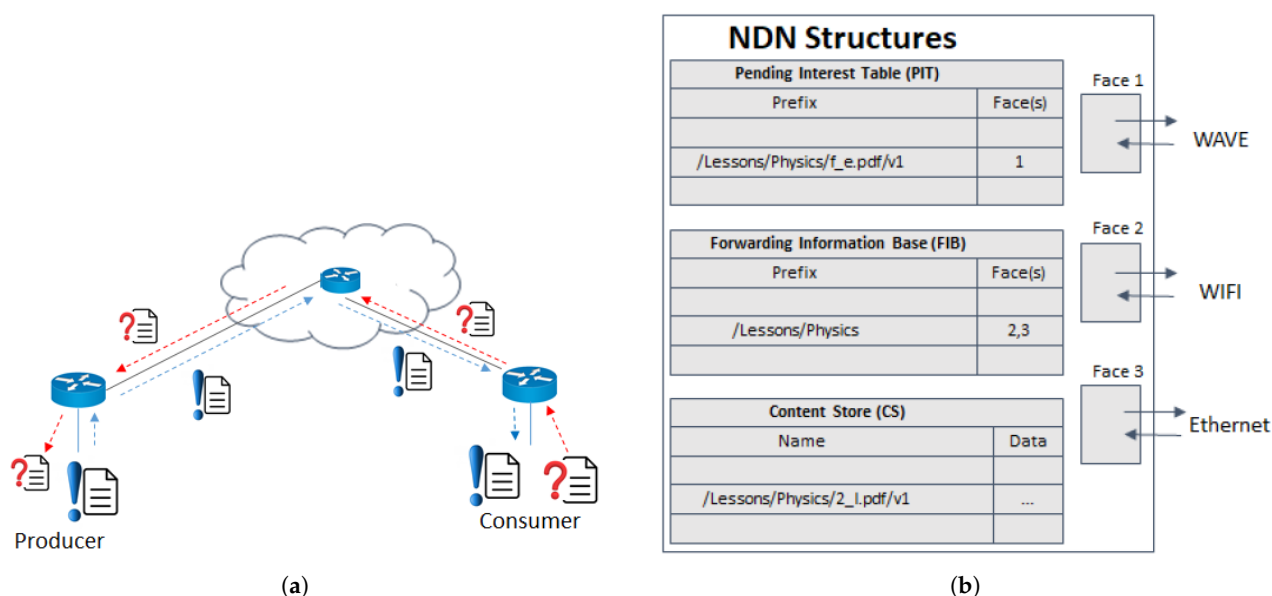


Figure 2. NDN basic principle. (a) Consumer and Producer interaction. (b) Node's control structures.

In short, in Named Data Networks the contents are independent of their physical location, such as IP addresses, being accessible from anywhere on the network. In addition, caching mechanisms speed up their access by distributing and keeping copies in the network nodes. The forwarding and routing mechanisms of NDNs make the nodes completely independent, not requiring the previous formation of networks, clusters or platoons for communication to flow between them. These characteristics better meet requirements such as scalability, heterogeneity and dynamicity that are increasingly present in new applications for metropolitan networks.

Yet, despite all the expected benefits of solutions that address the limitations of Internet protocols, due to their widespread use, it is not expected to be simply phased out, but a gradual replacement over time [24]. In this direction, our option was to design an architecture that allows peaceful coexistence between the IP-based network and the NDN solution. For a better understanding of such solution for coexistence and shared use of devices, below we give an overview of the city-scale communication infrastructure in the ATCLL.

3.2. Aveiro Tech City Living Lab Infrastructure

Aveiro Tech City Living Lab (ATCLL) is an advanced, large-scale sensing, communications and computation infrastructure, spread throughout the city of Aveiro in Portugal. The ATCLL is composed of an advanced communications infrastructure and a platform for the analysis and management of urban data that, together, make it possible to provide an open and large-scale technological laboratory in the city. The access infrastructure is supported by state-of-the-art fiber technology, reconfigurable radio units, 5G-NR radio and 5G network services, aggregating and interconnecting a range of sensors and remote information collection units that span the entire area of the city. Buses and garbage collection vehicles have also been equipped with communication units with Wi-Fi, ITS-G5 and 5G technologies, mobility sensors (GPS, traffic radars, LiDARs, and video cameras) and environmental sensors (such as temperature, humidity, pollution). This set of sensors record mobility and environmental data, building a complete live map of these parameters in the city, and providing the required data for its management, such as traffic monitoring and safe driving systems.

As illustrated in the left side of Figure 3, network nodes are also installed in the vehicles and local boats to transmit mobility and environment data to the data processing centre. In the vehicular network supported by ATCLL, the vehicles are equipped with On-Board Units (OBUs) to establish a connection with the Road Side Units (RSUs) and with the other vehicles. OBUs and RSUs were implemented using PC Engines APUs (<https://www.pcengines.ch/apu.htm> accessed on 7 June 2022) (Photo on the right side of the Figure 4) using an Ubuntu Linux distribution, offering vehicle-to-vehicle and vehicle-to-infrastructure communication over IEEE 802.11p interfaces (WAVE with ITS-G5 support), Wi-Fi, Ethernet and cellular communication (ready to integrate 5G modules). APUs use CPU AMD GX-412TC with 4 cores, 4GB DRAM, 30GB HDD, and Debian GNU/Linux 11.

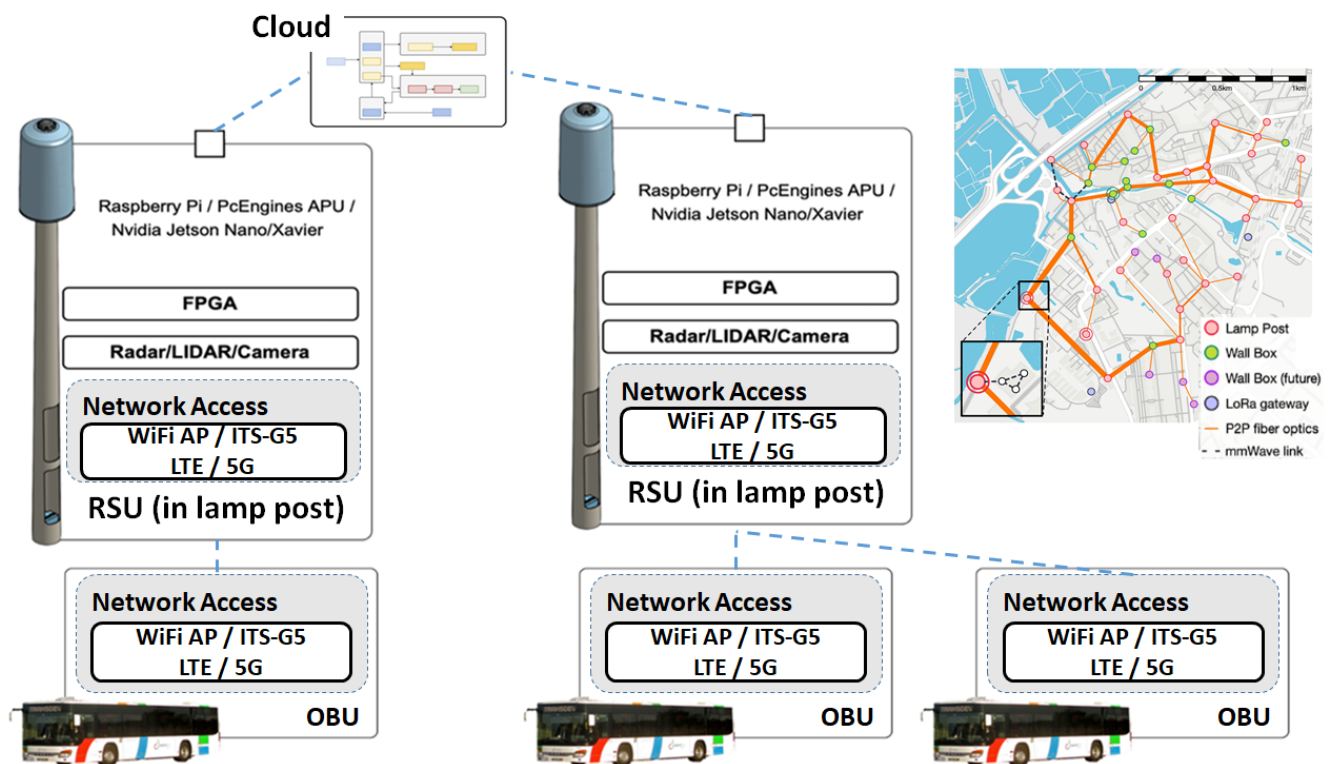


Figure 3. Aveiro Tech City Living Lab: communication architecture (on the left) and infrastructure's map (on the right).

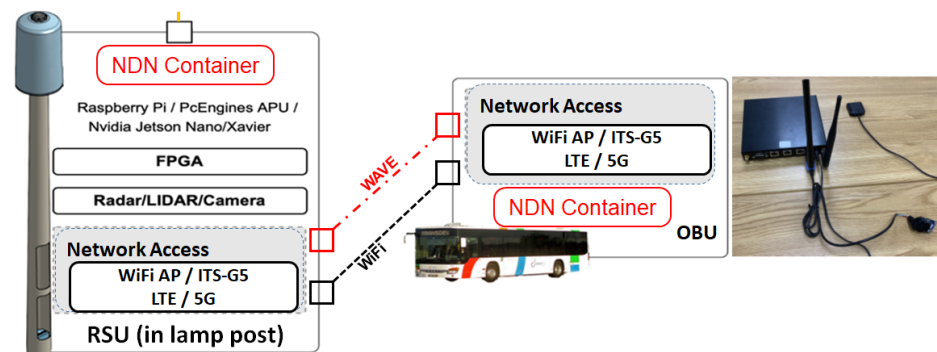


Figure 4. Deployment of NDN over ATCLL mobile communication units.

With restricted devices such as APUs, a scheme that allows dynamic publication of services on demand becomes essential. Thus, our solution for the deployment of NDN adopts a strategy of microservices made available through lightweight containers. This strategy allows us not only to deploy on demand, but also to include services migration to meet the mobility of OBUs, in addition to facilitating the configuration of devices on-the-fly as needed by the applications. Moreover, using lightweight containers, we isolate the NDN network stack from the TCP/IP network stack, ensuring complete independence from the operation of the networks while preparing the ATCLL for new network paradigms that may arise in the future.

To implement our NDN using containers we opted for Docker (<https://www.docker.com/> accessed on 7 June 2022). Docker is a platform for service virtualization where containers operate in isolation using their own software, libraries and configuration files. We prepare our Docker base image with S.O. Ubuntu (<https://ubuntu.com> accessed on 7 June 2022) where we install our NDN implementation. From this base image, we created a container on each node (RSU/OBU), highlighted in red in Figure 4, and configured them to communicate with each other, both by IEEE 802.11 Wi-Fi and by ITS-G5, in order to constitute a parallel and independent network.

3.3. NDN in the ATCLL

Our NDN solution was specially designed for IoT infrastructures taking advantage of mobility prediction information. In our project, we developed improvements regarding the Forwarding [3], Caching and Energy-efficiency [23], as well as exploring mobility-aware prediction capabilities [4]. For all these developments, we have used the NDN implementation available in the ndnSIM network simulator, an open source NS-3-based NDN simulator [25]. The ndnSIM offers efficient features for simulating the device faces of the NDN nodes. However, these mechanisms do not work in real communication devices interfaces. In order to appraise our solution in a mobile-aware environment as is the case of the ATCLL infrastructure, first we had to port our adjustments to an adequate format to be installed in constrained mobile devices. In this way, we performed this port in a modular approach, with the aforementioned changes to Forwarding and Caching processes being implemented in a Single-Data request format at the current time, with mobility-awareness capabilities, while the publish-subscribe capabilities should be included in following versions.

3.3.1. NDN Configuration

The NDN daemon is in charge of performing all the necessary setup of the NDN stack modules that implement the NDN architecture, such as forwarding and caching tables (PIT, CS, FIB) as well as verifying which physical interfaces exist on the device and installing an interface of communication between them and the NDN stack to allow for traffic routing using NDN packets as opposed to the commonly known IP. The components of these

modules can be configured via a control file which is analyzed by the daemon upon start, holding variables such as blacklisting of physical interfaces to be installed, among others.

Similarly, the NLSR routing protocol [10] behaves in analogous fashion, being in charge of calculating routing tables for each node in the infrastructure, thus, establishing their neighbourhood and guaranteeing that every Interest packet arriving at the edge of the network via the wireless medium can be properly forwarded towards their destination through a suitable NDN face, which can be controlled upon initialization via a control file. While all nodes in the network that wish to maintain NDN communication must have the NDN daemon installed and running, only fixed nodes composing the infrastructure are required to have the NLSR protocol running in order to route NDN traffic, due to their initial static neighbourhoods. Moreover, only certain nodes will act as either Producers or Consumers. This behavior is introduced via applications, which are designed specifically with a goal in mind, such as asking for a specific file, and must include proper packet creation in the NDN format and successful connection to the NDN daemon installed on the node. These applications are in charge of locally managing the content to send back to Consumers, acting as a Producer, and manage the Content received, when acting as a Consumer, using the daemon as a process to communicate with the outside world.

In its current state, the NDN platform does not intrinsically transmit traffic in the physical medium; instead, its network layer protocol makes use of the existing foundations set by IP which includes physical channels such as Ethernet wires and logical connections such as UDP or TCP tunnels over the existing Internet (Figure 5). To this extent, all communications between two direct nodes, as is the case with edge nodes and routers in the infrastructure, use UDP tunnelling to successfully route data (Interest or DATA packets) amongst them. On the other hand, when using the wireless medium, mainly between edge and mobile nodes, communications will take effect in a broadcast manner, in which all nodes will receive and transmit data without being directly influenced by its neighbours as expected from the case under study.

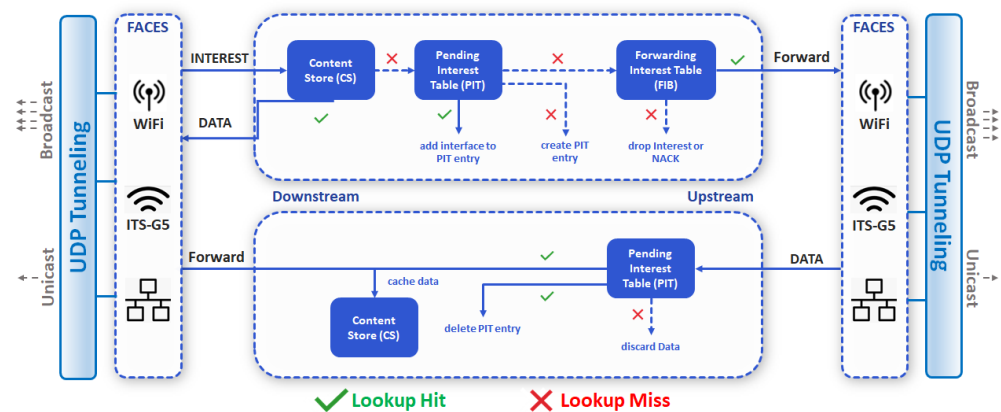
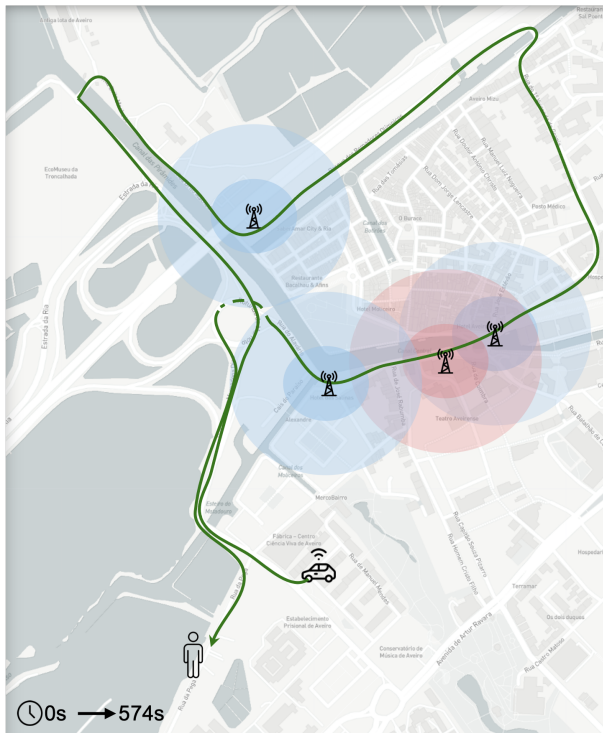


Figure 5. Forwarding process.

Figure 6 details the messages exchanged between a mobile Consumer and a mobile Producer, and the operations performed on each network element, when different radio access technologies are used.

3.3.2. Migration and Scalability

The adjustments mentioned previously result in a device capable of routing NDN traffic. In an ever-changing network environment, as is the case of our study, it is important that these capabilities can be replicated to various other devices in the network. This was achieved by the creation of a Docker image running the bare-minimum components in order to run the solution, which can be transferred seamlessly to other devices running the Docker daemon on-the-fly.



(a)



(b)



(c)



(d)

Figure 7. Route performed by the mobile node in the first experiment. (a) Between $t = 0\text{ s}$ and $t = 574\text{ s}$ the vehicle connects with four RSUs. (b) Between $t = 574\text{ s}$ and $t = 1000\text{ s}$ both vehicle and pedestrian are in communication range. (c) Between $t = 1000\text{ s}$ and $t = 1534\text{ s}$ the vehicle travels around the city, connecting with three RSUs. (d) Finally, the vehicle ends its journey next to the pedestrian.

Table 1 refers to the parameters considered in this scenario. Six files were considered with increasing size values which would be transmitted through ITS-G5 on the wireless section of the network. A single file will be under dissemination at a given time, meaning that only after the previous file is successfully downloaded, the next one will start its transmission. This approach aims to test two aspects: how the architecture behaves when multiple files are being delivered in consecutive manner, and the impact of the mobility in the delivery of large files, requiring the connection to multiple points of access to the infrastructure.

Table 1. City scenario parameters.

| | | | |
|----------------------------|-------------|-------------|-------------------------|
| Experiment Duration | | 1700 s | |
| Interest Timeout | | 4s | |
| Chunk Size | | 1490 | |
| Communication Type | | ITS-G5 | |
| Content | Type | Size | Number of Chunks |
| | .docx | 16 KB | 11 |
| | .jpg | 36 KB | 23 |
| | .jpg | 68 KB | 46 |
| | .jpg | 104 KB | 71 |
| | .pdf | 8.2 MB | 5760 |

The performance of the NDN solution in this scenario is discussed using the following indicators:

- **Interest satisfaction ratio:** measures the number of Interest packets arriving at the Producer, and how many of the Data packets, sent upon the reception of the Interest packet, arrive at the Consumer. This helps us to understand if the heuristics set on the Forwarding module, responsible for decisions about the forwarding of Interest and Data packets, are properly defined;
- **Network overhead:** a mobile network node moving at uncertain speed and at an unpredictable rate of contact with the network has a direct influence on the number of packets exchanged with the network mostly due to the inevitable packet loss that will occur, increasing the need for Interest retransmission. The number of Interests and Data packets traversing the network will be considered in this indicator;
- **Network contact time:** despite the strategic placement of RSUs, due to the unknown behavior of the vehicle traffic and weather conditions, it is inevitable that disconnections will occur. To ascertain the efficiency of content forwarding when contact is available, and the impact this may have in the results, we verify how long a mobile node is connected to the network, in seconds, throughout the experience.

Figure 8 presents the raw number of unique content chunks received by both types of mobile nodes in the network-vehicle and pedestrian. By interlacing the jumps in the amount of data received with the layout of the edge nodes in the network, we can observe the first itinerary of the vehicle (between 100 and 450 s in Figure 8a). Following this initial behavior, we see that the vehicle stops receiving content for a while, while the pedestrian starts to receive content, despite being isolated from the infrastructure, which proves that both mobile nodes made contact and the latter received the data through cache's vehicle. Then, the vehicle entering the second part of the circuit received the exclusive additional data, while the pedestrian retrieved all the additional content available in the second cycle. In general, we see that, although the vehicle could not retrieve the full extent of the available files, it was able to properly download data from RSUs placed around the city and deliver its content to other nodes via caching mechanisms.

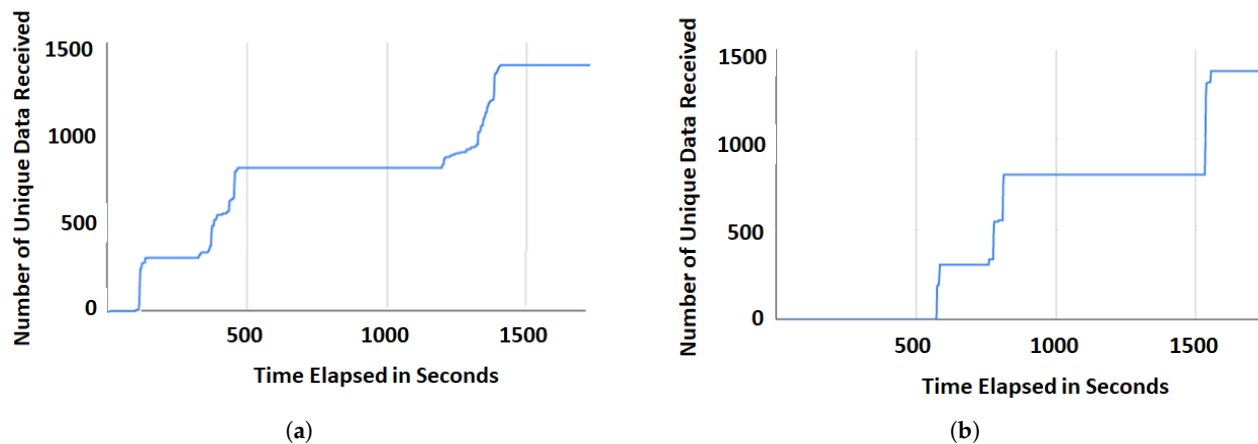


Figure 8. Number of unique data (chunks) received during the experiment. (a) Data downloaded by the vehicle. (b) Data downloaded by the pedestrian.

Figure 9a presents an overview of the number of packets (Interests and Data) transmitted and received by the vehicle. On the one hand, the number of packets shown by the red (satisfied Interests) and yellow (sent Data) bars shows that few retransmissions occurred when contact with the network was possible. On the other hand, the blue bar shows the small amount of replicated data due to the vehicle's simultaneous contact with several RSUs, mainly in the central part of the city where three edge nodes coexist. The results related to the mobile node representing the pedestrian were unconsidered as they took full advantage of the connection time to the vehicle, downloading all available cached contents with minimal retransmissions. Any Interest packets that resulted in timeouts and subsequent retransmission by the mobile nodes occur due to absence of contact with the network as opposed to issues regarding efficient Forwarding of the content through available paths. This can be addressed by adding more points of contact to the network.

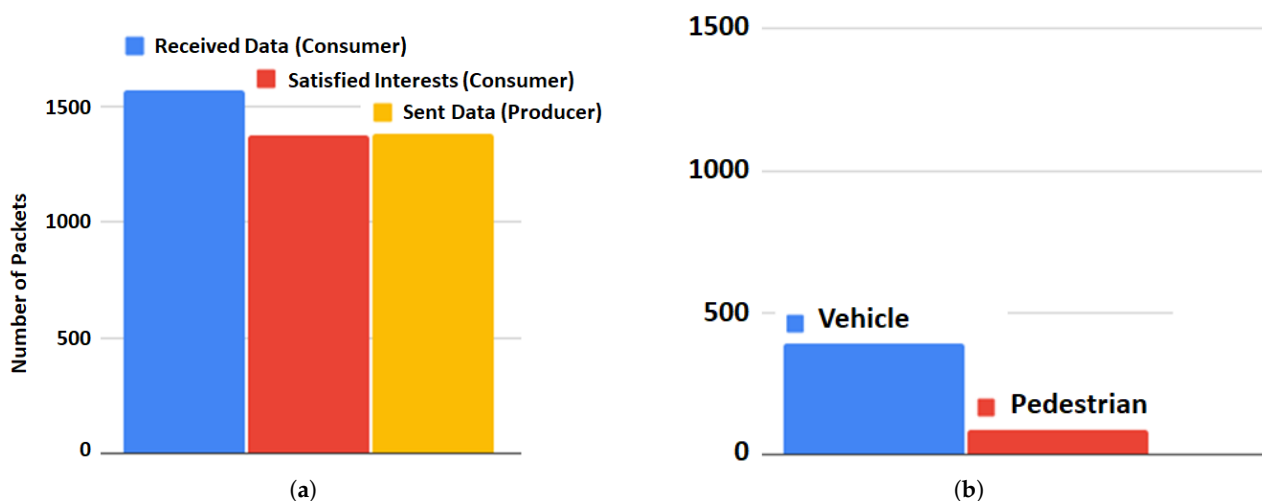


Figure 9. Network overhead. (a) Total amount of packets transmitted/received by the vehicle. (b) Total contact time between each mobile node and the infrastructure.

Figure 9b depicts the period of time each mobile node was successfully connected to the infrastructure, and thus able to send and receive content, compared to the total time elapsed. We can conclude that part of the reason the mobile nodes were not able to completely transfer all intended files was due to the reduced contact time with any cache node, totalling 26% for the vehicle (442 s over 1700 s) and 6% for the pedestrian (192 s over 1700 s), which only had the other mobile node as a source of content. Therefore, it

can be implied that increasing the amount of contact time with the network could directly improve the Interest satisfaction ratio.

4.2. NDN Using Multiple Radio Wireless Interfaces

A vehicular network presents a number of connectivity challenges that can be improved by adding extra points of contact across the city. Therefore, we decided to evaluate the behavior of the NDN using ITS-G5 and Wi-Fi simultaneously in ad-hoc mode. These tests were executed in a small-scale environment, following the same network elements used in the previous experiment (OBUs and RSUs), although without mobility. The goal was to evaluate the behavior of the NDN when two or more wireless communication technologies are available. In this evaluation there was the transmission of a single file, divided in several chunks, throughout the testbed.

To further evaluate the impact of different application behaviors we explore two paradigms: pipelined and non-pipelined. In a pipelined scenario all chunks will be requested by the Consumer at once, virtually injecting a great mass of packets in the network in an effort to quickly retrieve the data. In a non-pipelined scenario the content chunks will be requested sequentially, with new chunks only being requested upon successful retrieval of the previous ones, hence guaranteeing a slower yet reliable delivery of the file requested.

The test itself was performed by splitting the request of a 6 MB sized file through both wireless communication technologies, meaning that each will be in charge of retrieving half of the content chunks of the file (Figure 10). Furthermore, we considered both Unicast and Multicast communication methods to further enrich the study of the behavior of both technologies. In this scenario the performance of the NDN solution is assessed by measuring the number of packets transmitted (Interests) and received (Data), the number of timeouts, and the average delay per chunk which refers to the time between the creation of an Interest packet for a given chunk and the successful retrieval of its corresponding Data, taking into account all necessary retransmissions.

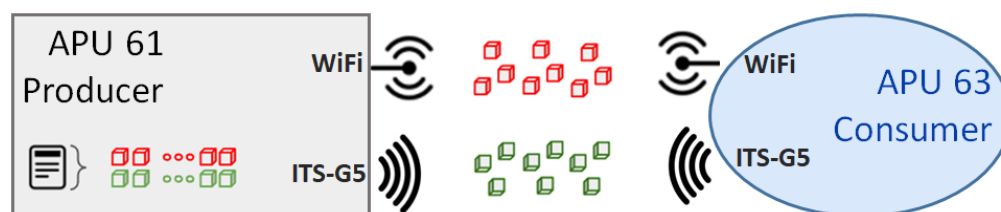


Figure 10. NDN in multiple wireless interfaces—requesting a 6 MB file using both ITS-G5 and Wi-Fi.

We started our tests by transmitting the entire file using only one of the technologies to have a baseline when managing the use of two technologies simultaneously. Table 2 shows the results on distributing the 6 MB file (1403 chunks for MTU = 4470).

Table 2. Non-pipelined results—full 6 MB file using ITS-G5 or Wi-Fi.

| ITS-G5 | | | | | Wi-Fi | | | |
|-----------|-----------------------------------|---------|--------|-----------|-----------------------------------|---------|--------|-----------|
| Multicast | Time | OutInts | InData | TimedOuts | Time | OutInts | InData | TimedOuts |
| | 94 s | 1412 | 1403 | 9 | 303 s | 1440 | 1403 | 37 |
| | Average delay per chunk: 27.89 ms | | | | Average delay per chunk: 57.07 ms | | | |
| Unicast | Time | OutInts | InData | TimedOuts | Time | OutInts | InData | TimedOuts |
| | 25 s | 1403 | 1403 | 0 | 82 s | 1403 | 1403 | 0 |
| | Average delay per chunk: 17.87 ms | | | | Average delay per chunk: 58.09 ms | | | |

Having established the performance benchmark for each technology, we split the file in half to be transmitted by one of the technologies so that we can assess the combined performance. Table 3 details the results obtained using the non-pipelined application for each technology.

Table 3. Non-Pipelined results—full 6 MB file using both ITS-G5 and Wi-Fi (3 MB to each interface).

| | ITS-G5 | | | | Wi-Fi | | | |
|------------------|-----------------------------------|---------|--------|-----------|-----------------------------------|---------|--------|-----------|
| | Time | OutInts | InData | TimedOuts | Time | OutInts | InData | TimedOuts |
| Multicast | 69 s | 719 | 711 | 8 | 149 s | 729 | 711 | 18 |
| | Average delay per chunk: 28.16 ms | | | | Average delay per chunk: 57.37 ms | | | |
| Unicast | 14 s | 711 | 711 | 0 | 42 s | 711 | 711 | 0 |
| | Average delay per chunk: 18.41 ms | | | | Average delay per chunk: 58.73 ms | | | |

The summary of the results when both technologies are used simultaneously is shown in Table 4. Firstly, we can verify that using multicast brings about several delays in the delivery of the full sized file, no matter which technology is considered, in part due to the use of the shared wireless medium to forward data, which results in timeouts. Secondly, it is notable that using unicast methodology presents much better quality of results, both in terms of reliability and speed of content delivery, and allows us to conclude that ITS-G5 is much faster in delivering content as is to be expected. When we consider both technologies to share efforts in requesting a file to the network, improvements are noticeable in all aspects. On the one hand, by sharing the wireless medium in different bands it alleviates the collision problem, which implies greater reliability in the delivery of packets. On the other hand, by sharing the burden over two different physical interfaces we can minimize the occurrence of issues caused by the forwarding of several packets simultaneously, as is noticeable by the reduced number of packets attended by each of them. Ultimately, the delivery of the full sized file over two different technologies presented better results over using solely Wi-Fi and worse results over using solely ITS-G5, although these values may change when congestion starts to occur in the network due to the increased number of nodes taking part in the wireless medium communications.

Table 4. Total download time using multi-technology and non-pipelined application.

| | Time | OutInts | InData | TimedOuts | Average Delay per Chunk |
|------------------|-------|---------|--------|-----------|-------------------------|
| Multicast | 149 s | 1448 | 1422 | 26 | 42.76 ms |
| Unicast | 42 s | 1422 | 1422 | 0 | 38.57 ms |

Tables 5 and 6 present the results regarding the use of the pipelined application in which all content requests are sent in bulk. We identify a significantly worse general performance when using a single communication technology, both under the unicast or multicast methods. Multicast methods suffered several Interest timeouts, due to the insufferable congestion introduced into the wireless medium, that resulted in increased periodicity to deliver the intended file. However, by using both Wi-Fi and ITS-G5 to forward the content, we were able to obtain much better performance under the form of reduced interest timeouts which ultimately resulted in lower times to retrieved the file successfully.

Table 5. Pipelined results—full 6 MB file using both ITS-G5 and Wi-Fi (3 MB to each interface).

| | ITS-G5 | | | | Wi-Fi | | | |
|------------------|-------------------------------------|---------|--------|-----------|-------------------------------------|---------|--------|-----------|
| | Time | OutInts | InData | TimedOuts | Time | OutInts | InData | TimedOuts |
| Multicast | 331 s | 27,441 | 711 | 26,730 | 212 s | 11,857 | 711 | 11,146 |
| | Average delay per chunk: 516.93 ms | | | | Average delay per chunk: 2957.79 ms | | | |
| Unicast | 19 s | 1551 | 711 | 840 | 37 s | 2650 | 711 | 1939 |
| | Average delay per chunk: 1644.91 ms | | | | Average delay per chunk: 3019.83 ms | | | |

Despite this increase in performance, it is clear that trying to retrieve content in a bulk, as is the case of the pipelined application, can be detrimental to the behavior of the network, and thus presents significant challenges to overcome which are not entirely related to the architecture itself, but instead are affected by it.

Table 6. Total download time using multi-technology and pipelined application.

| | Time | OutInts | InData | TimedOuts | Average Delay per Chunk |
|------------------|-------|---------|--------|-----------|-------------------------|
| Multicast | 331 s | 39,298 | 1422 | 37,876 | 516.93 ms |
| Unicast | 37 s | 4201 | 4266 | 2779 | 2197.48 ms |

The results show that there are grounds to believe that the use of several communication technologies to forward content simultaneously is beneficial to the well fare of architectures using NDN communications in mobile networks.

4.3. NDN According to the Network Characteristics

Over the course of our tests we have noticed that certain network conditions could have a significant impact in the NDN protocol operation, specially in the packet forwarding over the wireless medium. Such conditions can, in some cases, lead to a higher number of congestion situations. To this extent, we have performed a study on how different network characteristics affect the behavior of the NDN in an effort to further ascertain how better they can be utilized to minimize any negative consequences. The network parameters under study in this scenario are: (i) the Interest timeout, which represents the amount of time that a given node will wait for the corresponding Data packet to arrive; and (ii) the Maximum Transmission Unit (MTU), and consequently, the chunk size. We ran this experiment in the same laboratory environment used in the previous scenario. The main goal was the transmission of a 6 MB file from the infrastructure to the mobile node using wireless technologies, in an effort to obtain the lowest download time possible in a reliable manner. Finally, the indicators used to assess the performance of the NDN in this scenario are as follows:

- **Download duration:** measures the total time needed to download the Content starting from the moment that the first chunk is requested;
- **Number of timeout events:** every time a Consumer sends an Interest a timer is set, indicating the maximum amount of time available for the Data packet to arrive. If the timer expires, the Consumer retransmits the Interest packet, therefore increasing the network overhead.

Figure 11a shows that increasing the timeout value slightly can bring benefits in the time needed to download the file, but can also be detrimental or present no significant results as this value is increased to higher values. It is also clear that the NDN packet size has notable improvements: by simply doubling the size of the packet we were able to download the file 60% faster. Improvements to download time stagnate as we approach higher values to this metric, due to the fact that it is easier for longer sized packet to be lost in the wireless channel. A safe value to assume as a *sweet-spot* for our scenario would be an Interest timeout value of 6 seconds and a 4470 chunk-sized NDN packet, achieving one of the smallest download times while still leaving a window for any transmission errors to occur.

These values can be further attested by verifying the number of occurred timeouts for the different network characteristics, as depicted in Figure 11b. As the number of Interest timeouts increases, the number of timeouts naturally decreases because more time is given for the data packet to be delivered. However, it also needs to be taken into account that higher Interest timeout values will result in delayed Interest retransmissions, if needed, and consequently longer download times. To this extent, it is emphasized that a middle term must be achieved for a proper and reliable behavior of the network altogether.

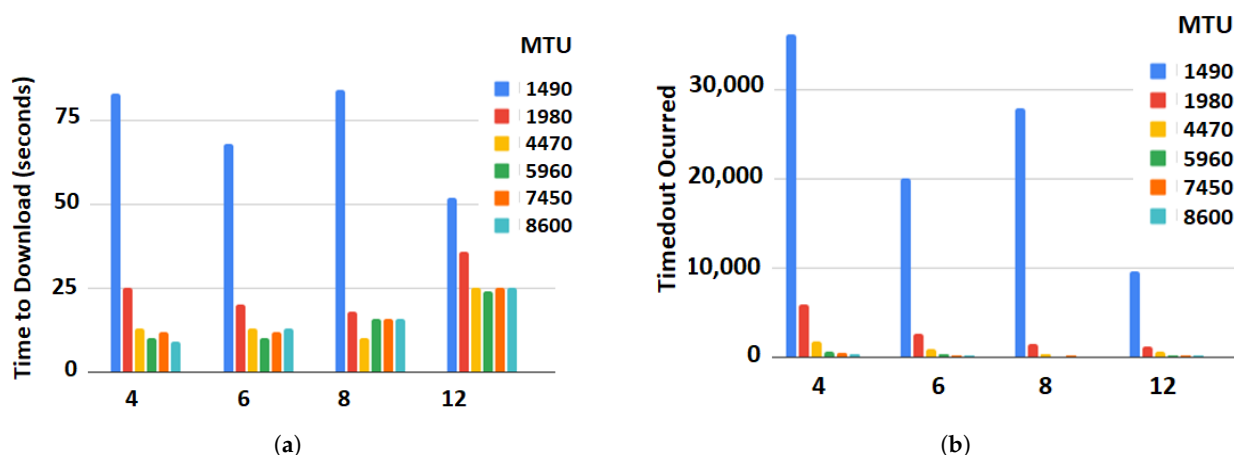


Figure 11. Download durations and number of timeout events for different MTUs and Interest timeouts. (a) Time to download file. (b) Timeouts.

5. Conclusions

In this article, we have discussed the implementation and performance of an NDN solution over a real smart city communication infrastructure. Our implementation, a container-based solution to be deployed in the network nodes, either mobile nodes or those belonging to the non-mobile infrastructure, was designed to allow the simultaneous operation of NDN and TCP/IP communications, offering facilities for on-the-fly network upgrades without compromising the network operation, and it allows the use and the performance evaluation of both communication paradigms simultaneously.

Evaluation tests, using real hardware equipment and mobility, have shown that our NDN solution, a modified version of the original NDN implementation with new heuristics about the forwarding, caching and support for publish-subscribe communications, is able to deliver the Content to the requested mobile users, directly from the infrastructure, or from the cache of other mobile nodes. In addition, we have shown that the NDN architecture can successfully explore the use of multiple communication interfaces for a faster and reliable Content retrieval. In the end, we analyzed how the performance of the NDN architecture is impacted by the network characteristics, experimenting different MTUs, that consequently lead to different chunk sizes.

As a future work, we aim to develop new heuristics for the management of the multiple communication interfaces by selecting the best interface for a given application according to the status of each interface. Another line of research includes the adaptation of the NDN solution to SDN-based environments and its assessment in the SDN-capable ATCLL communication infrastructure. Finally, to improve security in packet exchange, authentication mechanisms for NDN should be explored.

Author Contributions: Data curation, L.G.; Investigation, L.G., C.S. and M.L.; Supervision, C.S. and M.L.; Validation, L.G., C.S. and M.L.; Visualization, C.S. and M.L.; Writing—original draft preparation, L.G.; Writing—review and editing, C.S. and M.L. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the European Regional Development Fund (FEDER), through the Competitiveness and Internationalization Operational Programme (COMPETE 2020) of the Portugal 2020, Regional Operational Program of Lisbon (FEDER) and Foundation for Science and Technology, project InfoCent-IoT (POCI-01-0145-FEDER-030433), and by FCT/MEC through national funds and when applicable co-funded by FEDER – PT2020 partnership agreement under the project UIDB/50008/2020-UIDP/50008/2020.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Acknowledgments: We would like to thank Gustavo Amaral for the support on the experimentation.

Conflicts of Interest: The authors declare no conflict of interest

References

- Ahlgren, B.; Dannewitz, C.; Imbrenda, C.; Kutscher, D.; Ohlman, B. A survey of information-centric networking. *IEEE Commun. Mag.* **2012**, *50*, 26–36. <https://doi.org/10.1109/MCOM.2012.6231276>.
- Zhang, L.; Afanasyev, A.; Burke, J.; Jacobson, V.; claffy, k.; Crowley, P.; Papadopoulos, C.; Wang, L.; Zhang, B. Named Data Networking. *SIGCOMM Comput. Commun. Rev.* **2014**, *44*, 66–73. <https://doi.org/10.1145/2656877.2656887>.
- Gameiro, L.; Senna, C.; Luís, M. ndnIoT-FC: IoT Devices as First-Class Traffic in Name Data Networks. *Future Internet* **2020**, *12*, 207. <https://doi.org/10.3390/fi12110207>.
- Hernandez, D.; Gameiro, L.; Senna, C.; Luís, M.; Sargento, S. Handling Producer and Consumer Mobility in IoT Publish-Subscribe Named Data Networks. *IEEE Internet Things J.* **2022**, *9*, 868–884. <https://doi.org/10.1109/JIOT.2021.3069054>.
- Aboodi, A.; Wan, T.C.; Sodhy, G.C. Survey on the Incorporation of NDN/CCN in IoT. *IEEE Access* **2019**, *7*, 71827–71858. <https://doi.org/10.1109/ACCESS.2019.2919534>.
- Jacobson, V.; Smetters, D.K.; Thornton, J.D.; Plass, M.F.; Briggs, N.H.; Braynard, R.L. Networking Named Content. In Proceedings of the 5th International Conference on Emerging Networking Experiments and Technologies, Rome, Italy, 1–4 December 2009; ACM: New York, NY, USA, 2009; pp. 1–12. <https://doi.org/10.1145/1658939.1658941>.
- Jacobson, V.; Smetters, D.K.; Thornton, J.D.; Plass, M.; Briggs, N.; Braynard, R. Networking Named Content. *Commun. ACM* **2012**, *55*, 117–124. <https://doi.org/10.1145/2063176.2063204>.
- Ioannou, A.; Coileain, F.O.; Collins, D.; Zhang, Y.; Chen, B. CLONE: An NDN Architecture for Content Distribution at Remote Tourist Sites—A TCP/IP and NDN Comparison. In Proceedings of the Proceedings of the 5th ACM Conference on Information-Centric Networking, Boston, MA, USA, 21–23 September 2018; Association for Computing Machinery: New York, NY, USA, 2018; pp. 212–213. <https://doi.org/10.1145/3267955.3269018>.
- Guimarães, P.H.V.; Ferraz, L.H.G.; Torres, J.V.; Mattos, D.M.F.; Andrés F., M.P.; Andreoni L., M.E.; Alvarenga, I.D.; Rodrigues, C.S.C.; Duarte, O.C.M.B. Experimenting Content-Centric Networks in the future internet testbed environment. In Proceedings of the 2013 IEEE International Conference on Communications Workshops (ICC), Budapest, Hungary, 9–13 June 2013; pp. 1383–1387. <https://doi.org/10.1109/ICCW.2013.6649453>.
- Wang, L.; Lehman, V.; Mahmudul Hoque, A.K.M.; Zhang, B.; Yu, Y.; Zhang, L. A Secure Link State Routing Protocol for NDN. *IEEE Access* **2018**, *6*, 10470–10482.
- Mick, T.; Tourani, R.; Misra, S. LAsER: Lightweight Authentication and Secured Routing for NDN IoT in Smart Cities. *IEEE Internet Things J.* **2018**, *5*, 755–764. <https://doi.org/10.1109/JIOT.2017.2725238>.
- Papadopoulos, C.; Shannigrahi, S.; Afanasyev, A. In-Vehicle Networking with NDN. In Proceedings of the 8th ACM Conference on Information-Centric Networking, Paris, France, 22–24 September 2021; Association for Computing Machinery: New York, NY, USA, 2021; pp. 127–129.
- De Sena, Y.A.B.L.; Dias, K.L. Native versus Overlay-based NDN over Wi-Fi 6 for the Internet of Vehicles. In *Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering*; Springer: Cham, Switzerland, 2021; Volume 424. https://doi.org/10.1007/978-3-030-97124-3_5.
- Abane, A.; Daoui, M.; Bouzeffrane, S.; Muhlethaler, P. NDN-over-ZigBee: A ZigBee support for Named Data Networking. *Future Gener. Comput. Syst.* **2019**, *93*, 792–798. <https://doi.org/10.1016/j.future.2017.09.053>.
- Yang, W.; Wu, F.; Tian, K. High Performance Adaptive Video Streaming Using NDN WLAN Multicast. In Proceedings of the 8th ACM Conference on Information-Centric Networking, Paris, France, 22–24 September 2021; Association for Computing Machinery: New York, NY, USA, 2021; pp. 42–51.
- Wu, F.; Yang, W.; Ren, J.; Lyu, F.; Ding, X.; Zhang, Y. Adaptive Video Streaming Using Dynamic NDN Multicast in WLAN. In Proceedings of the IEEE INFOCOM 2020—IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS), Toronto, ON, Canada, 6–9 July 2020; pp. 97–102. <https://doi.org/10.1109/INFOCOMWKSHPS50562.2020.9162662>.
- Li, M.; Pei, D.; Zhang, X.; Zhang, B.; Xu, K. NDN Live Video Broadcasting over Wireless LAN. In Proceedings of the 2015 24th International Conference on Computer Communication and Networks (ICCCN), Las Vegas, NV, USA, 3–6 August 2015; pp. 1–7. <https://doi.org/10.1109/ICCCN.2015.7288434>.
- Li, M.; Pei, D.; Zhang, X.; Zhang, B.; Xu, K. Interest-suppression-based NDN live video broadcasting over wireless LAN. *Front. Comput. Sci.* **2017**, *11*, 675–687. <https://doi.org/10.1007/s11704-016-5563-x>.
- Ghasemi, C.; Yousefi, H.; Zhang, B. Internet-Scale Video Streaming over NDN. *IEEE Netw.* **2021**, *35*, 174–180. <https://doi.org/10.1109/MNET.121.1900574>.
- Bouk, S.H.; Ahmed, S.H.; Kim, D.; Song, H. Named-Data-Networking-Based ITS for Smart Cities. *IEEE Commun. Mag.* **2017**, *55*, 105–111. <https://doi.org/10.1109/MCOM.2017.1600230CM>.
- Rainer, B.; Petschornig, S. Challenges and Opportunities of Named Data Networking in Vehicle-To-Everything Communication: A Review. *Information* **2018**, *9*, 264. <https://doi.org/10.3390/info9110264>.

22. Nour, B.; Sharif, K.; Li, F.; Biswas, S.; Mounsla, H.; Guizani, M.; Wang, Y. A survey of Internet of Things communication using ICN: A use case perspective. *Comput. Commun.* **2019**, *142–143*, 95–123. <https://doi.org/10.1016/j.comcom.2019.05.010>.
23. Marques, D.; Senna, C.; Luís, M. Forwarding in Energy-Constrained Wireless Information Centric Networks. *Sensors* **2022**, *22*, 1438. <https://doi.org/10.3390/s22041438>.
24. Conti, M.; Gangwal, A.; Hassan, M.; Lal, C.; Losiouk, E. The Road Ahead for Networking: A Survey on ICN-IP Coexistence Solutions. *IEEE Commun. Surv. Tutor.* **2020**, *22*, 2104–2129. <https://doi.org/10.1109/COMST.2020.2994526>.
25. Mastorakis, S.; Afanasyev, A.; Zhang, L. On the Evolution of NdnSIM: An Open-Source Simulator for NDN Experimentation. *SIGCOMM Comput. Commun. Rev.* **2017**, *47*, 19–33. <https://doi.org/10.1145/3138808.3138812>.