

UNIVERSITÉ AMMAR THLEDJI LAGHOUAT

Faculté de Science

Département de Mathématiques et d'Informatique



THÈSE DE MASTER

*Pour obtenir diplôme Master Réseaux, Systèmes et Applications Réparties*

---

# Message forwarding in Vehicular Named Data Networking VNDN

---

*présenté par :*

BOUDELAA AMEL

*Encadré par :*

M. LAGRAA NASREDDINE

M. KERRACHE CHAKER

ABDELAZIZ

*Soutenu devant le jury composé de :*

M. GUELLOUMA Younes

M. BOUSBAA Fatima zahra

2016/2017

## *Résumé*

L'architecture Named Data Networking (NDN)[1] est une architecture entièrement nouvelle, issue du projet antérieur Content Centric Networking (CCN)[12], et représente une instance pionnière d'ICN. Elle est capable de capitaliser sur les points forts et les faiblesses de l'architecture internet actuelle [ref] pour s'adapter naturellement aux nouveaux modes de communication.

Les environnements sans fil remettent en question la capacité de TCP/IP à supporter les communications de bout en bout, en raison du manque de contrôle centralisé, la mobilité des nœuds, la topologie fortement dynamique, et la connectivité intermittente. En effet, les solutions que propose NDN à ces questions peuvent offrir un grand potentiel pour résoudre certains des problèmes rencontrés par les protocoles basés sur IP dans les réseaux sans fil.

Récemment, NDN a été adapté aux réseaux ad hoc de véhicules (VANET) que nous appelons l'architecture résultante Vehicular Named Data Networking (VNDN). Les travaux existants montrent que NDN apporte une amélioration significative de la performance pour les applications basées sur les contenus et peut être aussi bien adapté aux environnements mobiles qui caractérisent les réseaux VANET.

Depuis son l'apparition de VNDN, plusieurs designs ont été proposées dans ce travail nous allons présenter et discuter quatre conceptions VNDN : (1) Vehicular Named Data Networking VNDN[17], (2) Navigo [19], (3) Hierarchical Vehicular Named Data Networking HVNDN [21], (4) Vehicular Named Data NETworking VENDNET[22].

Pour évaluer la performance des stratégies de transfert de paquets NDN dans les scénarios urbains, nous illustrons les résultats expérimentaux menés par ndnSIM des stratégies suivantes : Best-Route ; Multidiffusion ; NCC ; Et Client-control.

# Abstract

Named Data Networking (NDN)[1] has been recently added to the future Internet family. It is basically an extension to the Content Centric Network (CCN) [12] that gives favorable promises in distributed wireless environments. The latter ones seriously call into question the capability of TCP/IP to support stable end-to-end communications due to lack of centralized control, nodes' mobility, dynamic topologies, intermittent connectivity, and harsh signal propagation conditions. Relying on name-based forwarding and in-network data caching, the NDN architecture has great potential to solve some of the problems encountered by IP-based protocols in wireless networks.

Recently, NDN has been adapted into Vehicular Ad hoc Networks (VANET) and hence, got the name Vehicular NDN (VNDN). Existing works show that NDN brings significant performance improvement for typical content-centric applications and can also fit the mobile environment well. However, directly applying NDN to Vehicular Ad hoc Networks (VANET) is confronted with great challenges such as consumer/provider mobility, Interest/Data forwarding, content caching and so on.

Several VNDN designs have been proposed to face these challenges and improve the performance of NDN packet forwarding especially in urban VANET scenarios. In this work, we are going to present four VNDN designs.

The first one is dubbed V-NDN [17], it illustrates NDN's promising potential in providing a unifying architecture that enables networking among all computing devices independent from whether they are connected through wired infrastructure, or ad hoc mode.

The second design discussed is Navigo [19], It takes a radically new approach to address the challenges of frequent disconnections and sudden topology changes in a vehicle network. Instead of forwarding packets to a specific moving car, Navigo aims at fetching specific pieces of data from multiple potential carriers of the data. The design provides : (1) a mechanism to bind NDN data names to the producer's geographic area(s); (2) an algorithm to guide Interests towards data producers using a specialized shortest path over the road topology; and (3) an adaptive discovery and selection mechanism that can identify the best data source across multiple geographic areas, as well as quickly react to changes in the V2X network.

The third design discussed is a prototype implementation of a hierarchical architecture for content distribution in vehicular network[21]. A cluster-based organization of vehicles is considered based on the Named Data Networking (NDN) model where a subset of reference vehicles named barycenters plays the NDN router role. Complementary functionalities of both short-range and cellular communications are used, the former for vehicle-to-vehicle (V2V), the latter for vehicle-to-infrastructure (V2I) communications, to off-load the fixed NDN infrastructure, and increase the chance of finding a content in the VANET, while limiting the number of broadcasted information.

And for last, the fourth design presented is VEHicular Named Data NETwork (VENDNET)[22] according to three different vehicle communication mechanisms, which are vehicle-to-infrastructure (V2I), a hybrid of vehicle to road side unit (V2R) and vehicle to vehicle (V2V).

To evaluate the performance of NDN packet forwarding strategies in urban scenarios, we illustrate the experimental results conducted by ndnSIM of the following strategies : Best-Route ; multicast ; NCC ; and client control.

## *Remerciement*

*La construction de ce mémoire n'aurait été possible sans l'intervention de certaines personnes. Qu'elles trouvent ici l'expression de mes plus sincères remerciements pour leurs précieux conseils.*

*La première personne que je tiens à remercier est mon encadrant Mr. (LAGRAA Nasreddine), pour l'orientation, la confiance, la patience qui ont constitué un apport considérable sans lequel ce travail n'aurait pas pu être mené au bon port. Qu'il trouve dans ce travail un hommage vivant à sa haute personnalité.*

*Mes remerciements s'étendent également à Mr. (KERRACHE Chaker Abdelaziz) pour son accueil, son soutien, et ses ses bonnes explications qui m'ont été précieuses dans l'accomplissement de ce modeste travail.*

*Mes vifs remerciements vont également aux membres du jury pour l'intérêt qu'ils ont porté à ce travail en acceptant de l'examiner et de l'enrichir par leurs propositions. Enfin ,Je tiens à exprimer mes sincères remerciements à tous les professeurs qui m'ont enseigné et qui par leurs compétences m'ont soutenu dans la poursuite de mes études.*



*Au nom d'Allah le plus grand merci lui revient de m'avoir guidé vers le droit chemin, de m'avoir aidé tout au long de mes années d'étude, de m'avoir donné le courage et la volonté de mener à terme ce présent travail.*

*A mes chers parents : Tous les mots du monde ne sauraient exprimer l'immense amour que je vous porte, c'est à travers vos encouragements et vos critiques que je me suis réalisée, j'espère avoir répondu aux espoirs que vous m'avez fondé en moi. Que dieu tout puissant vous garde et procure santé, bonheur, et longue vie.*

*A mes chers grands-parents : Puisse dieu vous protéger du mal et vous procurer une longue vie pleine de bonheur.*

*A la mémoire de ma chère tante Fatima : Le destin ne nous a pas laissé le temps pour jouir ce bonheur ensemble, puisse dieu tout puissant vous accueille dans son saint paradis.*

*A mes très chères sœurs Manel, Lina, Sabrin et Dalal et mon petit frère Mohammed Adel puisse l'amour et la fraternité nous unissent à jamais je vous souhaite la réussite et tout le bonheur qu'il faut pour vous combler.*

*A toute ma famille*

*A mes très chères amies Souad, Bouchra, et Hayet que notre amitié reste à jamais.*

*A tous ceux qui me sont chères. A tous ceux qui m'aiment. A tous ceux que j'aime. Je dédie ce mémoire.*

*BOUDELAA Amel*



# Table des matières

<b>Résumé</b>	<b>ii</b>
<b>Remerciement</b>	<b>v</b>
<b>1 Introduction générale</b>	<b>2</b>
<b>2 Préliminaires sur les réseaux Ad hoc de véhicules VANET</b>	<b>5</b>
2.1 Introduction	5
2.2 Les Réseaux VANET (Vehicular Ad-Hoc Network)	5
2.3 Caractéristiques des réseaux VANET	6
2.3.1 Le potentiel énergétique	6
2.3.2 L'environnement de communication et le modèle de mobilité	7
2.3.3 Topologie du réseau et mobilité	7
2.3.4 Connectivité du réseau	7
2.4 Les applications VANET	7
2.4.1 Les applications de gestion de trafic [3]	8
2.4.2 Les applications de confort et divertissement [3] [4]	8
2.5 Les types de communication dans les réseaux VANET	9
2.5.1 Communications Véhicule à Véhicule (V2V) [5]	9
2.5.2 Communication Véhicule à Infrastructure (V2I) [5]	9
2.5.3 Communication hybride [5]	9
2.6 Standards de communication dans les VANET	10
2.7 Les axes de recherche dans les VANET	14
2.7.1 Grille véhiculaire (VANET Grid - VGRID)	14
2.7.2 Le Cloud computing véhiculaire	14
2.7.3 Information Centric Networking (ICN) dans les réseaux VANET	15
2.8 Conclusion	16
<b>3 Named Data Networking : Vue d'ensemble</b>	<b>17</b>
3.1 Introduction	17
3.2 Architecture TCP/IP	17
3.3 Pourquoi faut-il trouver de nouvelles architectures Internet ?	19

3.3.1	L'architecture TRAIID	19
3.3.2	L'architecture DONA	20
3.3.3	L'architecture NetInf	20
3.3.4	L'architecture ICN	20
3.4	Named Data Networking NDN	21
3.5	Les noms de données	23
3.6	Les paquets NDN	24
3.6.1	Paquet de requête - interest packet (IntPK)	25
3.6.2	Paquet de données – data packet (DataPK)	25
3.7	Architecture des routeurs NDN	26
3.7.1	Content store - CS	26
3.7.2	Pending Interest Table - PIT	26
3.7.3	Forwarding Information Base - FIB	26
3.8	Le principe de fonctionnement NDN	27
3.9	Les nouveaux réseaux NDN	30
3.10	Conclusion	32
<b>4</b>	<b>Vehicular Named Data Networking VNDN</b>	<b>33</b>
4.1	Introduction	33
4.2	Apports de NDN aux réseaux VANET	33
4.3	Le 1 <sup>er</sup> design VNDN Vehicular Named Data Networking	35
4.3.1	Architecture VNDN	36
4.3.1.1	NDN Daemon	36
4.3.1.2	NDN Local Faces	37
4.3.1.3	Network Faces	37
4.3.1.4	Link Adaptation Layer (LAL)	37
4.3.1.5	Location service	37
4.3.2	Amélioration de la diffusion Wi-Fi pour les communications V2V	38
4.3.3	Discussion	41
4.3.3.1	Avantages	41
4.3.3.2	Inconvénients	41
4.4	Le 2 <sup>ieme</sup> design : Navigo, Transfert de données en utilisant les positions géographiques	42
4.4.1	Dénomination des zones géographiques	43
4.4.2	Cartographie des noms de données	43
4.4.3	Architecture Navigo	44
4.4.4	l'intégration des informations géographiques dans la pile NDN	45
4.4.5	Gestion de la FIB	46

4.4.5.1	La taille de la FIB . . . . .	47
4.4.6	La couche Link Adaption Layer (LAL) . . . . .	47
4.4.6.1	Calcul du chemin le plus court . . . . .	47
4.4.7	Le processus de transfert de données . . . . .	48
4.4.8	Discussion . . . . .	51
4.4.8.1	Avantages . . . . .	51
4.4.8.2	Inconvénients . . . . .	51
4.5	Le 3 <sub>ieme</sub> design : Architecture VNDN hiérarchique et dynamique . . . . .	51
4.5.1	Architecture du réseau . . . . .	52
4.5.1.1	La couche 0 . . . . .	52
4.5.1.2	La couche 1 . . . . .	53
4.5.1.3	La couche 2 . . . . .	53
4.5.1.4	La couche 3 . . . . .	53
4.5.2	Élection des Barycenters . . . . .	53
4.5.2.1	La procédure d'initialisation . . . . .	53
4.5.2.2	La Réélection d'un Barycenter . . . . .	54
4.5.3	Architecture des Barycenters . . . . .	54
4.5.4	Modèle de communication hiérarchique . . . . .	55
4.5.4.1	Recherche intra-cluster . . . . .	56
4.5.4.2	Recherche inter-cluster . . . . .	56
4.5.4.3	Recherche dans les clusters des cellules distinctes . . . . .	57
4.5.4.4	Recherche dans Internet-NDN . . . . .	58
4.5.5	Discussion . . . . .	58
4.5.5.1	Avantages . . . . .	58
4.5.5.2	Inconvénients . . . . .	58
4.6	Le 4 <sub>ieme</sub> design : Vehicular Named Data Networking VEDNET . . . . .	59
4.6.1	le routage réactif dans NDN . . . . .	60
4.6.2	Exemple de l'architecture VEDNET simulé . . . . .	61
4.6.3	Discussion . . . . .	63
4.6.3.1	Avantage . . . . .	63
4.6.3.2	Inconvénients . . . . .	63
4.7	Conclusion . . . . .	63
<b>5</b>	<b>Résultats et discussion</b> . . . . .	<b>66</b>
5.1	Introduction . . . . .	66
5.2	Le simulateur Ns3 . . . . .	66
5.3	L'environnement de simulation ndnSIM . . . . .	67
5.3.1	La conception de ndnSIM . . . . .	67

5.3.1.1	ndn : :L3Protocol . . . . .	68
5.3.1.2	ndn : :Face . . . . .	68
5.3.1.3	ndn : :ContentStore . . . . .	69
5.3.1.4	ndn : :PIT . . . . .	69
5.3.1.5	ndn : :FIB . . . . .	69
5.3.1.6	ndn : :ForwardingStrategy . . . . .	70
5.4	Simulation et évaluations de la performance des stratégies de transmission . . . . .	70
5.4.1	La stratégie Best-Route . . . . .	70
5.4.2	La stratégie de contrôle des clients (Client Control Strategy) . . . . .	71
5.4.3	La stratégie NCC . . . . .	71
5.4.4	La stratégie Multicast . . . . .	71
5.4.5	Paramètres de simulation . . . . .	71
5.4.6	Les métriques de performance . . . . .	72
5.4.6.1	Le taux de délivrance des paquets (Packets Delivery Ratio - PDR) . . . . .	72
5.4.6.2	Le temps de livraison des données (data delivery time) . . . . .	72
5.4.6.3	Évaluation du taux de délivrance des paquets (PDR) . . . . .	72
5.4.6.4	Évaluation du temps de livraison des données . . . . .	74
5.5	Conclusion . . . . .	76
<b>6</b>	<b>Conclusion générale</b> . . . . .	<b>77</b>
6.1	Perspectifs . . . . .	78
	<b>Bibliographie</b> . . . . .	<b>79</b>

# Table des figures

2.1	Types de communications dans les réseaux VANET (V2V et V2I)[14]. . . . .	6
2.2	Les applications VANET. . . . .	8
2.3	Scénarios de communication dans le réseau VANET. . . . .	9
2.4	La pile protocolaire du standard WAVE.[6] . . . . .	11
2.5	La pile protocolaire ETSI.[7] . . . . .	11
2.6	Le Cloud computing véhiculaire . . . . .	15
3.1	L'architecture internet basée sur TCP/IP. . . . .	18
3.2	L'architecture de sablier (internet et NDN). [1] . . . . .	21
3.3	Named data Networking NDN.[1] . . . . .	22
3.4	Les noms hiérarchiques. . . . .	24
3.5	La structure des paquets.[1] . . . . .	25
3.6	L'architecture d'un routeur NDN. [1] . . . . .	27
3.7	Le fonctionnement de NDN. . . . .	28
3.8	Le processus exécuté par un routeur NDN lors de la réception d'un IntPK.[13]	29
3.9	Le processus exécuté par un routeur NDN lors de la réception d'un DataPK.[13] . . . . .	30
4.1	Architecture VNDN [17]. . . . .	35
4.2	Implémentation VNDN par[16]. . . . .	36
4.3	Schéma illustratif du processus de diffusion d'un IntPK [16]. . . . .	39
4.4	Processus de diffusion d'un IntPK. [16] . . . . .	40
4.5	La carte MGRS [18]. . . . .	43
4.6	Architecture VNDN-Navigo[18]. . . . .	44
4.7	Mappage GeoFaces – zone géographique[18]. . . . .	45
4.8	La stratégie de transfert de données adoptée par Navigo[18]. . . . .	46
4.9	Calcul du plus court chemin.[18] . . . . .	48
4.10	Le processus de transfert de données[19]. . . . .	50
4.11	Architecture hiérarchique VNDN à quatre couches[20]. . . . .	52
4.12	Les modules du Barycenter[20]. . . . .	55
4.13	Recherche intra-cluster[20]. . . . .	56
4.14	Recherche inter-cluster[20]. . . . .	57

4.15	Deux aspects de la communication des véhicules [21]. . . . .	59
4.16	Diagramme illustrant les opérations effectuées par un véhicule NDN [21]. . . . .	61
4.17	Architecture VEDNET[21]. . . . .	62
5.1	Les composants de ndnSIM [25]. . . . .	68
5.2	Abstraction de couche de communication dans les scénarios ndnSIM [25]. . . . .	69
5.3	Le PDR calculé en fonction du nombre de nœuds. . . . .	73
5.4	Le PDR calculé en fonction du nombre IntPK. . . . .	73
5.5	Le PDR calculé en fonction du nombre de producteurs. . . . .	74
5.6	Le temps de livraison de données calculé en fonction du nombre des nœuds. . . . .	75
5.7	Le temps de livraison de données calculé en fonction du nombre des IntPK (en microseconde). . . . .	75
5.8	Le temps de livraison de données calculé le nombre des producteurs (en microseconde). . . . .	76

# Liste des tableaux

2.1	la pile protocolaire du standard WAVE[7] . . . . .	12
2.2	La pile protocolaire du standard ETSI [7] . . . . .	13
5.1	Paramètres de simulation . . . . .	71

# Liste d'abréviations

<b>VANET</b>	<b>V</b> ehicular <b>A</b> d hoc <b>N</b> ETwork
<b>MANET</b>	<b>M</b> obile <b>A</b> d hoc <b>N</b> ETwork
<b>NDN</b>	<b>N</b> amed <b>D</b> ata <b>N</b> etworking
<b>CCN</b>	<b>C</b> ontent <b>C</b> entric <b>N</b> etworking
<b>ICN</b>	<b>I</b> nformation <b>C</b> entric <b>N</b> etworking
<b>VNDN</b>	<b>N</b> amed <b>D</b> ata <b>N</b> etworking
<b>RSU</b>	<b>R</b> oad <b>S</b> ide <b>U</b> nit
<b>ITS</b>	<b>I</b> ntelligent <b>T</b> ransportation <b>S</b> ystem
<b>IVC</b>	<b>C</b> ommunication <b>I</b> nter <b>V</b> ehicule
<b>WSN</b>	<b>W</b> ireless <b>S</b> ensor <b>N</b> etwork
<b>V2V</b>	<b>V</b> ehicule to <b>V</b> ehicule
<b>V2I</b>	<b>V</b> ehicule to <b>I</b> nfrastucture
<b>VGRID</b>	<b>V</b> ANET <b>G</b> rid
<b>VCC</b>	<b>V</b> ehicular <b>C</b> loud <b>C</b> omputing
<b>DONA</b>	<b>D</b> ata <b>O</b> riented <b>N</b> etworking <b>A</b> rchitecture
<b>PSIRP</b>	<b>P</b> ublish <b>S</b> ubscribe <b>I</b> nternet <b>R</b> outing <b>P</b> aradigm
<b>NetInf</b>	<b>N</b> etwork of <b>I</b> nformation
<b>DNS</b>	<b>D</b> omain <b>N</b> ame <b>S</b> ystem
<b>Ip</b>	<b>I</b> nternet <b>P</b> rotocol
<b>TCP</b>	<b>T</b> ransmission <b>C</b> ontrol <b>P</b> rotocol
<b>DOS</b>	<b>D</b> enial of <b>S</b> ervice
<b>DDOS</b>	<b>D</b> istributed <b>D</b> enial of <b>S</b> ervice
<b>IntPK</b>	<b>I</b> ntrest <b>P</b> acket
<b>DataPK</b>	<b>D</b> ata <b>P</b> acket
<b>PIT</b>	<b>P</b> ending <b>I</b> nterest <b>P</b> acket
<b>CS</b>	<b>C</b> ontent <b>S</b> tore
<b>FIB</b>	<b>F</b> orwarding <b>I</b> nformation <b>B</b> ase
<b>IOT</b>	<b>I</b> nternet <b>O</b> f <b>T</b> hings
<b>LAL</b>	<b>L</b> ink <b>A</b> daptation <b>L</b> ayer

<b>GPS</b>	<b>G</b> lobal <b>P</b> ositioning <b>S</b> ystem
<b>MGRS</b>	<b>M</b> ilitary <b>G</b> rid <b>S</b> ystem
<b>UTM</b>	<b>U</b> niversal <b>T</b> ransverse <b>M</b> ercator
<b>UPS</b>	<b>U</b> niversal <b>P</b> olar <b>S</b> tereographic



# Chapitre 1

## Introduction générale

Au cours des dernières années, les réseaux véhiculaires ad hoc (VANET) ont intéressé les constructeurs d'automobiles, les chercheurs universitaires, les agences gouvernementales ainsi que d'autres organismes de normalisation. Par conséquent, de nombreux standards, applications et mécanismes de routage ont été proposés pour répondre aux spécificités de cette classe de réseaux. En effet, cette attention croissante aux réseaux VANET, est due principalement à leur impact social et économique profond sur le plan mondial, qui est lié notamment à la grande gamme d'applications conçues pour ces environnements véhiculaires, telles que les applications de sécurité et d'efficacité du transport, qui sont employées pour diminuer le risque d'accidents de voitures (surtout dans les zones urbaines), et pour optimiser également l'échange de flux entre véhicules, ainsi que de fournir des services de confort et de commodité aux voyageurs

Les réseaux VANET ont été prouvés fiables et favorables à plusieurs classes d'applications, telle que les applications de sécurité routière qui consistent à renforcer la prévention des accidents et à munir nos voitures et nos routes de capacités permettant de rendre la conduite plus sûre (les informations sur le trafic, les accidents, les dangers, les déviations possibles, les informations météorologiques, etc.). Une autre classe d'applications de ce concept consiste à améliorer le confort des passagers et rendre le temps passé sur les routes plus conviviale (accès à internet, jeux interactifs entre les passagers des véhicules proches, service de chat, aider les personnes à se suivre sur la route, etc.). Ces applications sont l'exemple type de ce que nous appelons les systèmes de transports intelligents (ITS, Intelligent Transportation System) et dont le but est d'améliorer la sécurité, l'efficacité et la convivialité dans les transports routiers à travers l'utilisation des nouvelles technologies de l'information et de la communication.

En effet, les VANET se rapprochent de jour en jour de la réalité dans la vie quotidienne en fournissant non seulement une grande capacité de calcul, mais aussi une variété d'interfaces sans fil telles que 3G/LTE, WiMAX, Wifi, IEEE 1901 (Power Line Communication) et 802.11p (DSRC / WAVE), etc. Idéalement, une voiture devrait pouvoir utiliser toutes ces interfaces pour communiquer avec les infrastructures routières ou avec d'autres véhicules dans le réseau, selon les besoins de ses applications. En effet chaque fois que plus

d'une interface est disponible, le véhicule devrait pouvoir choisir parmi eux la meilleure ou d'utiliser plusieurs en parallèle.

La notion de connexion de véhicules aux infrastructures routières existe depuis longtemps, mais peu de pays l'ont déployé. En outre, une variété de standards a été développée pour la communication directe véhicule à véhicule (V2V), mais l'utilisation reste toujours limitée à la communication à un saut en raison du problème de collision. Par ailleurs, Il y a eu de nombreuses publications explorant l'utilisation de V2V pour pouvoir supporter les applications à longue portée, mais malheureusement, la majorité des solutions proposées ne permettent pas de surmonter les limitations de TCP/IP.

En dépit de leur grande capacité de contribuer à la gestion du trafic et à l'augmentation de la sécurité des conducteurs, les VANETs nécessitent encore des efforts de recherche pour faire face à des défis techniques liés principalement à la topologie fortement dynamique, la connectivité intermittente, et la diversité spatio-temporelle etc. Cependant, l'architecture Internet classique basée sur la pile protocolaire TCP / IP ne peut pas gérer efficacement ces problèmes. Nous avons donc besoin d'un changement de paradigme dans les réseaux de véhicules afin de pouvoir relever les défis majeurs de la conception TCP/IP. Le paradigme Information Centric Networking (ICN) a été préconisé par un certain nombre de groupe de recherche qui travaillent sur les futures architectures d'Internet.

Named Data Networking (NDN) [1] est une architecture entièrement nouvelle, issue du projet antérieur Content Centric Networking (CCN)[12], et représente une instance pionnière d'ICN. Elle est capable de capitaliser sur les points forts et les faiblesses de l'architecture internet actuelle pour s'adapter naturellement aux nouveaux modes de communication.

L'avantage de NDN réside dans sa capacité de traiter les noms de données plutôt que les adresses IP des hôtes. Plus précisément, NDN permet aux nœuds de nommer les données requises au lieu de nommer les hôtes destinataires. Cela rend facile la communication entre les différentes entités du réseau sans aucun besoin de configuration a priori des paramètres (i.e : adresse IP, serveur de noms, etc).

Named Data est une bonne approche pour les communications véhiculaires, plus particulièrement pour les communications V2V [ref]. Elle peut être utile pour conquérir la connectivité intermittente qui est due principalement à la forte mobilité dans les VANET, ainsi d'accélérer la propagation des données dans le réseau tout en affaiblissant le problème de collision.

Dans ce travail, on s'intéresse à l'étude des architectures Vehicular Named Data Networking VNDN, et des stratégies de transmission (Forwarding stratégies) de NDN. Le mémoire est structuré en quatre chapitres organisés comme suit :

Le premier chapitre d'état de l'art a pour objectif de donner une vue d'ensemble des VANET. Nous allons présenter dans un premier temps les réseaux VANET en définissant

brièvement leurs principales caractéristiques, leurs différentes classes d'applications, ainsi que leurs modes et standards de communication. Et nous concluons ce chapitre par les défis à relever pour une transmission efficace de données.

Le deuxième chapitre intitulé Named Data Networking donne une explication approfondie de l'architecture NDN et son principe de fonctionnement.

Le troisième chapitre sera consacré aux différentes architectures Vehicular Named Data Networking (VNDN) qui résultent de la fusion des VANET et NDN.

Dans le dernier chapitre nous allons présenter les résultats et discussions des simulations des stratégies de transmission (Forwarding stratégies) de NDN.

Enfin, nous terminons ce mémoire par donner une conclusion générale qui résume notre travail.

## Chapitre 2

# Préliminaires sur les réseaux Ad hoc de véhicules VANET

### 2.1 Introduction

L'industrie du transport représente aujourd'hui un moteur économique très important et un facteur crucial dans le développement des sociétés. Devant ce progrès les pays constructeurs d'automobiles et les consortiums d'industriels étaient amenés à créer une nouvelle catégorie d'applications des réseaux sans fil dans laquelle les plateformes communicantes sont embarquées dans des véhicules aux fins de fournir une connectivité et des services divers aux conducteurs et aux passagers. Ces plateformes constituent alors ce que l'on appelle les réseaux véhiculaires VANET qu'on va les détailler dans la suite de ce chapitre.

### 2.2 Les Réseaux VANET (Vehicular Ad-Hoc Network)

Les réseaux ad hoc de véhicules (VANET) sont créés en appliquant les principes des réseaux mobiles ad hoc (MANET), pour permettre la communication au sein d'un groupe de véhicules à portée les uns des autres et entre les véhicules et les équipements fixes RSU (Road-Side Unit) se trouvant au voisinage, usuellement appelés équipements de la route [voir figure 2.1]. Grâce aux progrès des technologies sans fil les réseaux VANET constituent aujourd'hui une composante importante du système de transport intelligent (Intelligent Transportation System- ITS), qui fournit plusieurs catégories d'applications visant particulièrement l'augmentation de la sûreté des passagers, l'efficacité du trafic et l'info-divertissement. Dans ces réseaux, chaque véhicule est capable de gérer, d'organiser et de mieux exploiter les informations envoyées par les véhicules se trouvant dans son voisinage, ou par les RSUs. Les réseaux VANET ont la particularité d'avoir une topologie

de réseau qui change très vite et très fréquemment en raison de la forte mobilité des véhicules.

Il est à noter que le terme Vehicular Ad-Hoc Network ou VANET est devenu aujourd'hui le synonyme de la communication inter-véhicules (IVC), bien que l'accent reste encore mis sur l'utilisation des infrastructures comme RSUs ou réseaux cellulaires :

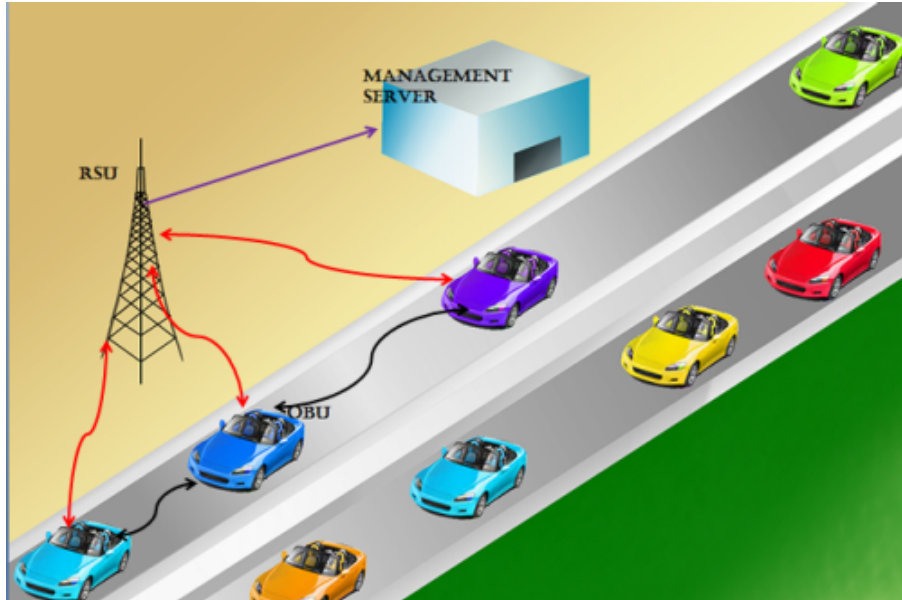


FIGURE 2.1 – Types de communications dans les réseaux VANET (V2V et V2I)[14].

## 2.3 Caractéristiques des réseaux VANET

Les réseaux VANET se distinguent des réseaux ad hoc traditionnels par un certain nombre de caractéristiques spécifiques dont on peut citer [2] [3] :

### 2.3.1 Le potentiel énergétique

Contrairement aux réseaux ad hoc traditionnels (wireless sensor network WSN par exemple) où la contrainte d'énergie représente un facteur limitant, les entités des réseaux véhiculaires disposent de suffisamment d'énergie qui peut alimenter les différents équipements électroniques d'un véhicule intelligent, ce qui permet une grande capacité de traitement ainsi que l'utilisation simultanée de plusieurs interfaces de communication.

### 2.3.2 L'environnement de communication et le modèle de mobilité

Les réseaux véhiculaires imposent la prise en compte d'une grande diversité environnementale (zone autoroutière, zone urbaine, zone rurale). Ils nécessitent également la prévention des conditions climatiques et les contraintes topologiques. Le déplacement des véhicules quant à lui est lié aux infrastructures routières soit dans des autoroutes ou au sein d'une zone métropolitaine.

Le modèle de mobilité peut être affecté par plusieurs facteurs, y compris : l'atténuation du signal, multi-path fading, les obstacles dans les zones urbaines, ainsi que le comportement des conducteurs, etc.

### 2.3.3 Topologie du réseau et mobilité

La vitesse de déplacement des véhicules peut provoquer des changements topologiques très fréquents. Cette dernière varie selon l'environnement de communication, elle est en moyenne de 50 km/h en zones urbaines et peut dépasser 120 km/h sur les autoroutes. Par conséquent, un véhicule peut rapidement rejoindre ou quitter le réseau en un temps très court, mais en revanche les mouvements des véhicules peuvent être prédictibles puisqu'ils se déplacent dans des trajectoires bien définies.

### 2.3.4 Connectivité du réseau

Le problème de partitionnement du réseau survient principalement lorsque la densité des nœuds est faible. Les véhicules se déplacent alors en groupes isolés non connectés. Il devient dans ce cas difficile voire impossible d'assurer les communications de bout-en-bout. A l'inverse, une forte densité de véhicules permet au réseau d'être connexe et donc il existe, généralement, au moins un chemin entre deux nœuds qui souhaitent communiquer.

## 2.4 Les applications VANET

Les applications VANET peuvent être classifiées en deux grandes catégories [voir figure 2.2] :

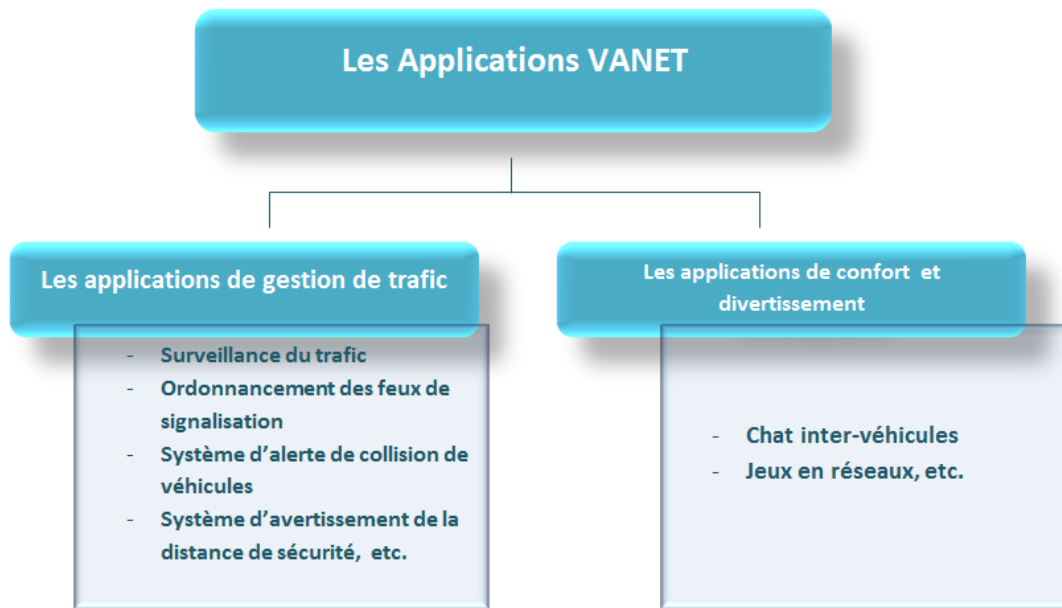


FIGURE 2.2 – Les applications VANET.

### 2.4.1 Les applications de gestion de trafic [3]

L'objectif principal de cette catégorie d'applications est la sécurité des conducteurs et les passagers ainsi que l'équilibrage de la circulation des véhicules sur les routes pour réduire l'embouteillage et les risques d'accidents. Ce qui peut diminuer par conséquent les pertes humaines, la durée des voyages et la consommation d'énergie.

Les véhicules et les RSU sont impliqués dans les opérations de surveillance des surfaces de la route, et des autres véhicules :

- Un Véhicule peut notifier, par l'envoi d'un message d'alerte, d'autres véhicules en cas d'accident, glissement de terrain, infraction de conduite, ou descente brusque, etc ;
- les données de trafic en temps réel peuvent être stockées dans la RSU et peuvent être mis à la disposition des véhicules. Cela peut jouer un rôle important dans la résolution des problèmes tels que les embouteillages, l'évitement de congestion et les alertes d'urgence telles que les accidents etc.

### 2.4.2 Les applications de confort et divertissement [3] [4]

L'objectif de ces applications est de rendre les voyages plus agréables en permettant aux passagers d'accéder à Internet ou à la messagerie et de télécharger des fichiers via une communication avec l'infrastructure, ou en exploitant les communications V2V pour réaliser des applications de chat inter-véhicule, des jeux en réseaux, etc.

## 2.5 Les types de communication dans les réseaux VANET

On distingue trois types de communication dans les réseaux véhiculaires dans la sécurité des réseaux sans fil véhiculaires [voir figure 2.3] :

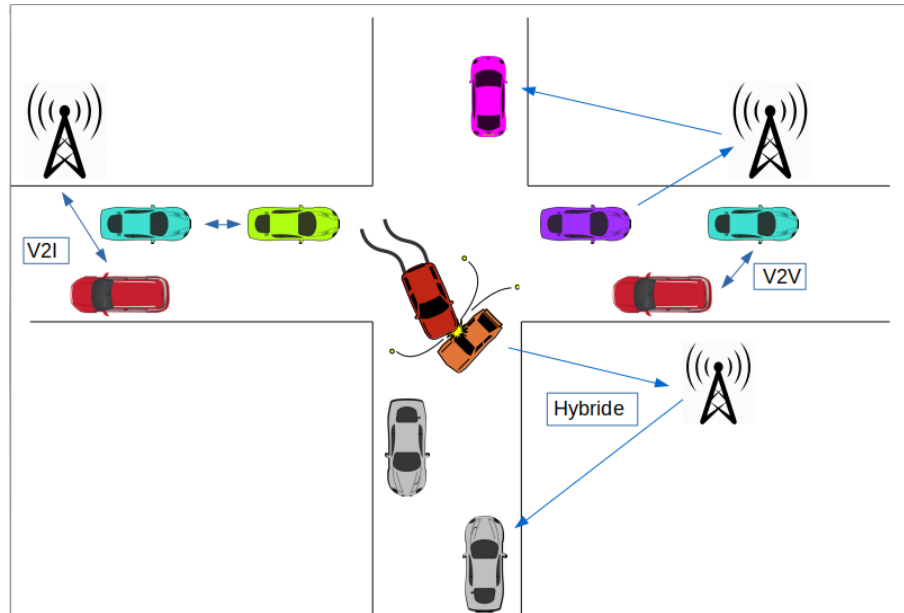


FIGURE 2.3 – Scénarios de communication dans le réseau VANET.

### 2.5.1 Communications Véhicule à Véhicule (V2V) [5]

Dans ce mode de communication, un véhicule peut communiquer directement avec un autre véhicule si ce dernier se situe dans sa zone radio, et cela sans s'appuyer sur une quelconque infrastructure ce qui permet de former une architecture décentralisée.

### 2.5.2 Communication Véhicule à Infrastructure (V2I) [5]

Dans ce mode de communication, les véhicules doivent se connecter à des stations fixes (RSUs) pour l'acquisition ou la transmission d'informations. Les RSUs sont connectées entre eux via des liaisons filaires et fournissent de multiples services tels que l'accès à Internet, l'échange de données de voiture-a-domicile, etc.

### 2.5.3 Communication hybride [5]

Ce mode de communication combine les deux modes V2V et V2I, en permettant à un véhicule d'échanger les informations avec les autres véhicules du réseau, ainsi qu'avec les

RSU de la route (l'échange d'informations avec les RSU peut se faire dans les deux sens, c'est-à-dire véhicule-RSU ou RSU-véhicule), même si les uns ne sont pas dans la portée des autres, et cela en utilisant d'autres véhicules comme étant des routeurs mobiles.

## 2.6 Standards de communication dans les VANET

Les efforts de standardisation des piles protocolaires des réseaux VANET sont réalisés aux États-Unis, en Europe et au Japon, au vu de leur dominance dans le domaine de l'industrie automobile. Toutes les piles protocolaires développées dans ces différents pays s'appuient sur la même technologie radio IEEE 802.11p WLAN, qui a été intégrée dans des normes plus larges telles que Wireless Access in Vehicular Environments – WAVE , Communications Access for Land Mobiles - CALM et Dedicated Short-Range Communications - DSRC [6].

Dedicated Short-Range Communications (DSRC) est une norme sans fil dédiée aux communications V2V et V2I de courte portée dans des emplacements précis. En 1999 la FCC (US Federal Communications Commission) a alloué suivant des études approfondies, 75 MHz du spectre dans la bande de fréquences de 5,9 GHz. Cette gamme de fréquence a été divisée en 7 canaux de 10MHz, dont un canal représente CCH (Control Channel) qui est utilisé pour diffuser les messages beacons et les messages d'alertes liés à la sécurité routière ainsi que pour annoncer les services et les applications offertes sur les canaux de service. Les six autres canaux représentent SCH (Service Channel) qui sont destinés aux transactions des différents services annoncés. Le standard WAVE : En 2003, le groupe de travail IEEE a développé le fameux IEEE 802.11p WAVE en s'appuyant sur la norme DSRC [voir figure 2.4] [6].

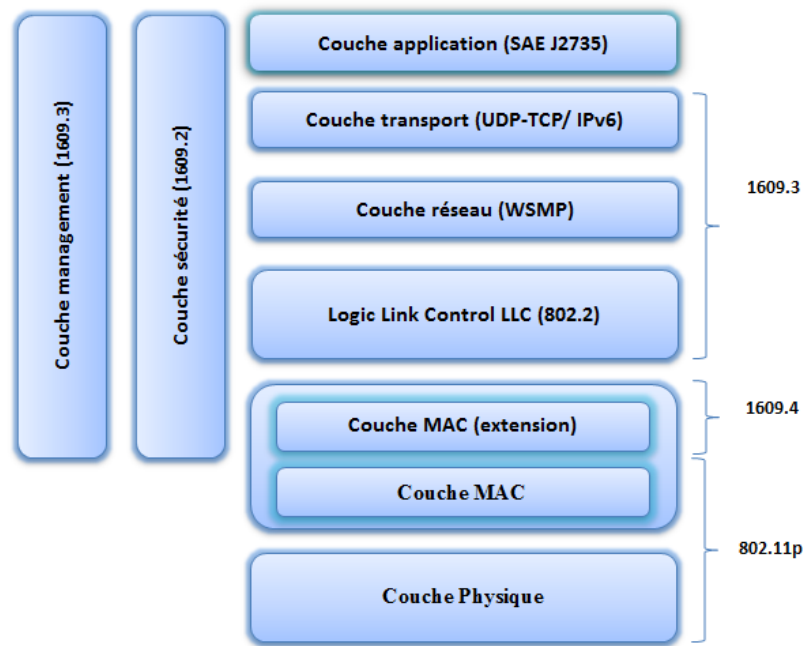


FIGURE 2.4 – La pile protocolaire du standard WAVE.[6]

En Europe, le standard ETSI ITS G5 [7], s'appuie sur une variante de la même technologie radio avec quelques adaptations qui œuvrent sur cinq canaux réservés dans la bande de fréquence de 5,9 GHz. La figure 2.5 illustre la pile protocolaire du standard ETSI :

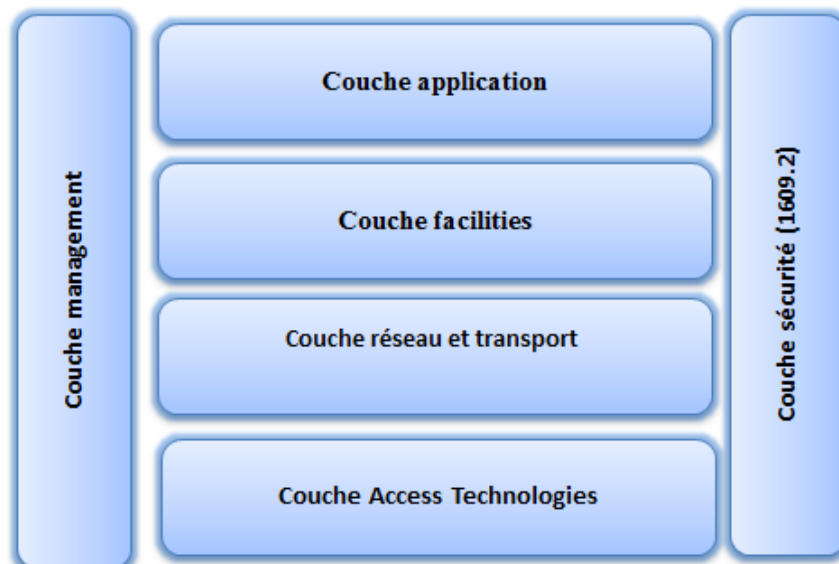


FIGURE 2.5 – La pile protocolaire ETSI.[7]

En comparant les deux standards on peut conclure que :

TABLE 2.1 – la pile protocolaire du standard WAVE[7]

Couche physique (PHY)	<ul style="list-style-type: none"> <li>• Utilise le standard IEEE 802.11p qui possède les mêmes fonctionnalités de la couche physique de la norme 802.11a ;</li> <li>• utilise la technique OFDM et pouvant supporter des débits de données différents allant jusqu'à 27 Mb/s (pour des distances jusqu'à 1000 mètres).</li> </ul>
Couche MAC	<ul style="list-style-type: none"> <li>• Utilise le standard IEEE 802.11p pour assurer l'accès au canal ;</li> <li>• utilise le principe de CSMA/CA (Carrier Sense Multiple Access with Collision Avoidance) ;</li> <li>• supporte la gestion de la qualité de service tout comme la norme IEEE 802.11e.</li> </ul>
Couche LLC	Basée sur le standards IEEE 802.2.
Couches réseau et transport	<p>Pour les services de niveau réseau et transport incluant l'adressage, le routage on utilise le standard IEEE 1609.3 (WAVE Networking Services) ; Le standard définit le message WSM (WAVE Short Message) et le protocole d'échange associé WSMP (WAVE Short Message Protocol) afin d'assurer la fonctionnalité du réseau et des couches de transport pour les applications liées à la sécurité ; Cette couche utilise le protocole IPv6 en raison de la taille réduite des en-têtes.</p>
Couche application	<p>Basée sur le standard SAE J2735 qui spécifie un ensemble de messages, tel que BSM (Basic Safety Message) ; Ce standard assure l'interopérabilité des applications DSRC.</p>
Le standard IEEE 1609.2 (cross layer)	Le standard IEEE 1609.2 décrit les services de sécurité en mettant l'accent sur le traitement et le format des messages sécurisés.
Le standard IEEE 1609.3 (cross layer)	Le standard 1609.3 fournit des fonctions de gestion du réseau WAVE, telles que le traitement des requêtes de service des couches supérieures, ainsi que l'affectation des canaux, la surveillance des annonces de services WAVE, et la configuration du protocole IPv6 à l'aide des données reçues à partir d'une autre source WAVE.

TABLE 2.2 – La pile protocolaire du standard ETSI [7]

La couche application	Elle gère les applications de la sécurité routière, divertissement etc.
La couche facilities	Fournit une ensemble de fonctions et des structures de données pour pouvoir stocker, agréger et conserver des données de différents types et sources et les rendre ainsi accessibles aux applications. En ce qui concerne la communication, la couche facilities fournit différents types de messages, tels que le message hello CAM (Cooperative Awareness Messages) et DENM (Decentralized Environmental Notification Messages).
La couche réseau et transport	La couche de réseau et de transport comprend des protocoles de transmission de données qui fournissent les fonctionnalités suivantes <ul style="list-style-type: none"> <li>• la livraison de données de bout en bout ;</li> <li>• le transfert fiable des données ;</li> <li>• le contrôle des flux et l'évitement des congestions.</li> </ul> Un autre protocole particulier est utilisé par la couche réseau, qui est l'IPv6.
La couche management	Elle est responsable de la configuration des stations, et de l'échange des informations entre les différentes couches.
La couche sécurité	Elle fournit des services de sécurité et de confidentialité aux différentes couches de la pile, tels que la gestion des identités et des informations de sécurité.
La couche Access Technologies	Elle est basée sur le standard IEEE 802.11p et comprend les différents supports de communication et protocoles connexes de la couche de liaison de données. Les technologies d'accès ne sont pas limitées à un type spécifique de média

- Les standards IEEE WAVE, et ETSI gèrent les applications de sécurité routière en utilisant les mêmes couches physique et MAC (IEEE 802.11p) ;
- les messages hello : l'ETSI utilise les messages CAM et DENM tandis que IEEE WAVE utilise le message BSM ;
- le standard ETSI intègre une nouvelle couche entre les couches transport et celle d'application appelée facilities.

## 2.7 Les axes de recherche dans les VANET

Dès leur apparition, les réseaux VANET s'imposent comme un domaine de recherche innovant, dans lequel les thèmes classiques des MANET (tels que le routage, la localisation, la préservation de vie privée, etc.) ont été repris.

Mais on atteste ces dernières années à l'émergence de nouvelles applications liées principalement à l'interaction possible avec d'autres domaines et au progrès significatif de l'industrie véhiculaire. Parmi ces nouvelles applications, on peut citer :

### 2.7.1 Grille véhiculaire (VANET Grid - VGRID)

Les grilles véhiculaires sont des systèmes de surveillance routière entièrement distribués, dans lesquels chaque véhicule partage des informations sur sa localisation et sur l'état de ses capteurs avec ses voisins, afin de modifier collectivement et dynamiquement leurs modèles de flux de trafic en temps réel. Contrairement aux grilles d'ordinateurs traditionnelles, dans VGrid[8], la topologie change avec le temps, cela permet de résoudre le problème de smoothing traffic flow, et peut aider ainsi à la localisation des accidents et à la diminution de congestion grâce au contrôle du trafic et l'échange des informations de localisation et de la vitesse en temps réel entre les véhicules via des messages d'alerte.

### 2.7.2 Le Cloud computing véhiculaire

Les véhicules et les capteurs situés dans une même zone locale peuvent générer et traiter des données de différents types. Ces données sont stockées, recherchées et consommées aux alentours de cette zone locale avant leur expiration. Récemment, les chercheurs ont introduit un nouveau modèle informatique qui exploite cette caractéristique des réseaux VANET, appelé le Cloud computing véhiculaire (VCC)[9][10]. Le VCC peut être défini comme un groupe de véhicules autonomes qui coopèrent à la détection, et à la communication des ressources physiques pour les coordonner et les attribuer dynamiquement à des utilisateurs autorisés, dans le but de résoudre les problèmes des systèmes centralisés. En effet, les conducteurs peuvent exploiter les ressources embarquées dans leurs véhicules et les attribuer temporairement aux autres véhicules qui demandent un service spécifique.

Par exemple, un conducteur peut envoyer une requête au Cloud véhiculaire en demandant l'état du trafic routier dans un segment quelconque [voir figure 2.6], pour répondre à cette requête, les véhicules qui sont enregistrés dans le système VCC, et ayant reçu la requête vont créer, maintenir et diffuser l'information au véhicule concerné. Dans ce cas, les véhicules vont aider à la résolution des incidents de circulation routière, la chose qui n'est pas possible avec les systèmes de gestion de la circulation centralisés. Le système VCC peut également ouvrir la porte à de nombreux nouveaux services véhiculaires tout en tenant en compte les caractéristiques du réseau VANET (ie : services multimédia, utilisation illimitée des ressources informatiques sur Internet et pour stocker ou télécharger des données depuis / vers Internet, etc.).

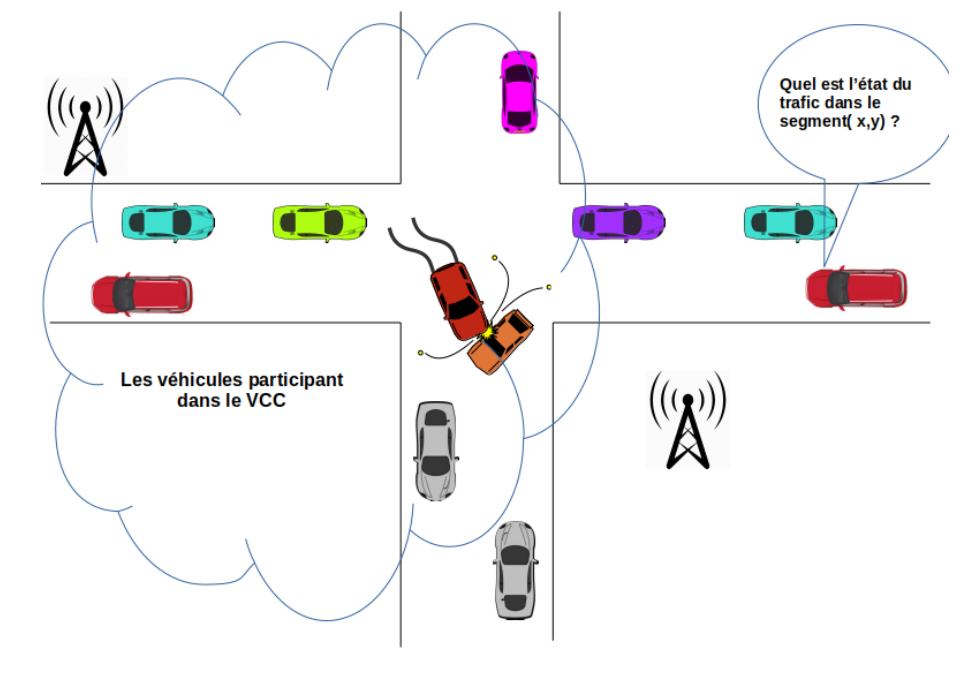


FIGURE 2.6 – Le Cloud computing véhiculaire

### 2.7.3 Information Centric Networking (ICN) dans les réseaux VANET

Information Centric Networking (ICN)[11] a été initialement conceptualisé comme une forme générale d'architecture de communication pour parvenir à une distribution de données efficace sur Internet. Les concepts principaux d'ICN visent à créer une architecture Internet centrée sur les données, plutôt que sur les hôtes, où les opérations principales sont effectuées sur le contenu plutôt que sur les adresses IP. Parmi les architectures récemment proposées pour l'ICN on peut citer notamment, DONA (Data-Oriented Network Architecture), PSIRP (Publish-Subscribe Internet Routing Paradigm), NetInf (Network of Information) et NDN (Named Data Networking).

NDN a été récemment étendu au réseau VANET pour conquérir les défis de mobilité et de connectivité intermittente qui sont difficiles à traiter dans l'architecture traditionnelle IP, et pour concevoir ainsi une distribution de contenu de faible latence, haute capacité et fiable dans les cadres de véhicules.

## 2.8 Conclusion

Les réseaux ad hoc de véhicules (VANET) sont un cas particulier des réseaux ad hoc mobiles, dont les entités communicantes sont embarquées dans des véhicules, et sont capables d'échanger les informations par voie radio dans le but d'améliorer la sécurité routière ou de permettre l'accès à Internet aux passagers. Dans ce chapitre, nous avons donné une vue globale sur les réseaux VANET et leurs principales caractéristiques ainsi que divers exemples d'applications. Dans le chapitre suivant nous introduisons l'architecture Named Data Networking (NDN).

## Chapitre 3

# Named Data Networking : Vue d'ensemble

### 3.1 Introduction

L'Internet actuel, qui a été conçu il y a plus de 40 ans, a évolué d'un champ d'expérimentation en une infrastructure commerciale ossifiée. Cependant la vitesse de cette transformation a représenté une grande frustration pour les architectes des réseaux, qui n'ont pas eu le temps d'élaborer une structure optimisée. Par conséquent, de nombreux problèmes techniques et non techniques sont apparus, (i.e. sécurité, mobilité, distribution des contenus, etc.). Comme les tentatives de correction de ces problèmes à travers des changements incrémentaux ont rendu l'architecture actuelle plus complexe, les chercheurs ont commencé à étudier de nouvelles architectures internet pour corriger ces défauts majeurs.

Named Data Networking NDN est une architecture entièrement nouvelle, qui est capable de capitaliser sur les points forts et les faiblesses de l'architecture actuelle afin de s'adapter naturellement aux nouveaux modes de communication. Elle hérite de la forme de sablier de l'architecture IP, mais elle remplace le modèle de livraison de données de bout en bout par un autre modèle qui est piloté par le récepteur. Dans la suite de ce chapitre, nous allons détailler l'architecture NDN et ses concepts. Mais avant tout ça, il est important de rappeler l'architecture IP.

### 3.2 Architecture TCP/IP

L'une des définitions historiques d'Internet a été donnée par Ed Kroll Le directeur du National Center for Supercomputing Applications, qui a résumé la nature pluridimensionnelle du réseau comme suit, Internet est [12] :

- Un réseau de réseaux fonctionnant avec le protocole TCP/IP ;
- une communauté de personnes qui utilisent et développent ces réseaux ;
- l'ensemble des ressources mises à disposition de ces communautés.

L'architecture internet actuellement utilisée correspond à une superposition de plusieurs couches dont les fonctions sont différentes. Ces couches fondamentales d'Internet sont liées au transport (infrastructures physiques), aux applications, et enfin aux informations échangées. Cette architecture est basée sur la spécification du protocole TCP/IP standard, conçue pour interconnecter deux réseaux, ce double protocole d'Internet assure une séparation entre les fonctions de transport et les fonctions de traitement des informations. Cette séparation est l'un des principes essentiels de l'Internet : le principe de bout en bout qui permet à n'importe quel nœud de communiquer avec n'importe quel autre, peu importe sa position dans le réseau.

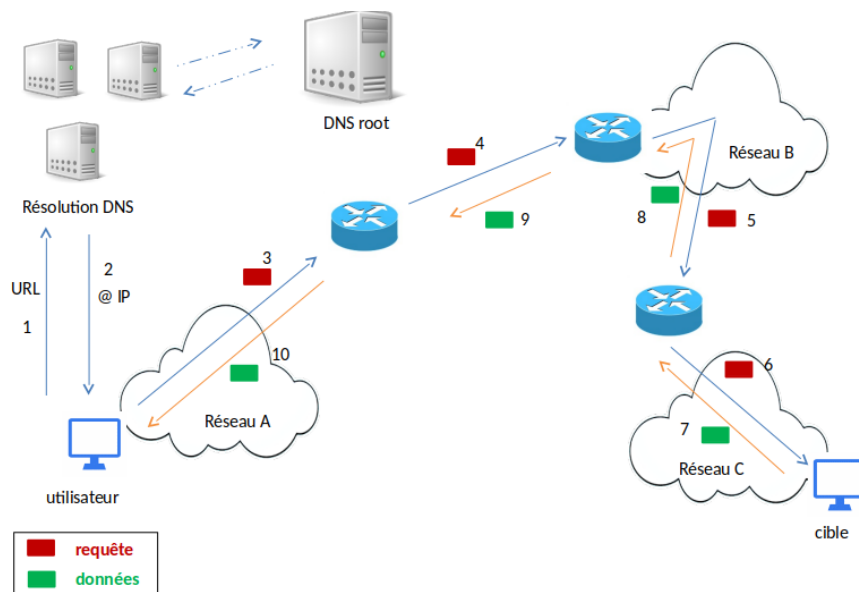


FIGURE 3.1 – L'architecture internet basée sur TCP/IP.

Lorsqu'un hôte sur Internet veut obtenir l'adresse IP d'un autre hôte en fonction de son nom de domaine :

- il envoie une requête au serveur de noms local (DNS)[figure 3.1]. Si le serveur de noms local est en mesure de répondre à cette requête, en utilisant les informations qui les possède dans sa base de données, il renvoie directement l'adresse IP de l'hôte cible ;
- en effet, la plupart du temps, le serveur DNS local est bien peu savant et demande à un autre serveur de lui donner la réponse. Par ailleurs, chaque serveur DNS étant responsable d'un domaine ou d'un petit nombre de domaines, la résolution consiste alors d'aller chercher l'information sur les serveurs DNS de manière hiérarchique, jusqu'à trouver le serveur de noms approprié pour l'hôte cible, ensuite la réponse sera transférée à l'hôte initial suivant le chemin inverse de la résolution ;

- une fois avoir l'adresse IP de l'hôte cible, l'utilisateur initial peut alors contacter ce dernier en envoyant un datagramme IP qui va se propager dans le réseau d'un routeur à autre, ou même d'un sous réseau à autre jusqu'à atteindre sa destination finale ;
- en réponse à la requête, l'hôte cible renvoie à son tour un autre datagramme IP qui peut suivre le même chemin initial ou encore il peut suivre un autre chemin alternatif pour arriver à l'utilisateur initial, cela dépend de l'état du réseau (collision, congestion, rupture de lien, etc) ;
- le fait de pouvoir suivre un chemin alternatif en retour à l'utilisateur initial est un avantage de l'architecture TCP/IP.

### 3.3 Pourquoi faut-il trouver de nouvelles architectures Internet ?

Malgré l'efficacité et la fiabilité du modèle de communication TCP/IP, de nombreux problèmes d'évolutivité sont apparus, on peut citer par exemple qu'il est devenu sujet aux attaques DOS/DDOS, et qu'il peut poser des dégradations de performance dans les petits réseaux avec des bandes passantes de tailles réduites, etc. En effet, toutes les tentatives de résolution de ces conflits ont rendu l'internet de plus en plus complexe et difficile à maintenir. De ce fait, les chercheurs ont commencé à étudier de nouvelles architectures telle que TRAIID, DONA (Data-Oriented Network Architecture), NetInf (Network of Information), et Information Centric Networking (ICN), etc, pour tenir compte des faiblesses de l'architecture actuelle et s'adapter aux nouveaux modes de communication.

#### 3.3.1 L'architecture TRAIID

L'un des premiers efforts de réalisation des réseaux centrés sur les données, est le projet TRAIID[11] qui a débuté en 1999 par l'Université de Stanford. La motivation principale des créateurs de TRAIID à l'époque, était de réduire le volume de trafic HTTP (qui représentait entre 70 et 80 % du trafic Internet), ainsi que de résoudre le problème d'insuffisance d'espace d'adressage IPv4. Afin d'atteindre cet objectif, ils ont essayé d'apporter les requêtes HTTP à la source la plus proche en créant un réseau de recouvrement basé sur IPV4 pour que la réponse soit obtenue dans le moins de temps possible. En effet, leur but principal était de notifier les routeurs de leurs données à l'aide des en-têtes des paquets échangés, pour améliorer le routage.

L'autre possibilité, notamment utilisée dans TRIAD (une autre version de TRAIID), est le routage par nom, mais à cause de l'effet intensif du routage IP et du fonctionnement TCP, cette méthode n'a pratiquement pas été utile. Les chercheurs qui ont proposé l'idée

de TRIAD ont affirmé que la nécessité de l'utilisation de IPV6 pourrait être éliminée en utilisant IPV4 et le NAT et aussi la mise en œuvre correcte de TRIAD.

TRIAD préconise également l'utilisation de noms DNS comme identificateurs globaux pour identifier de façon unique chaque hôte final.

### 3.3.2 L'architecture DONA

**DONA** [11] est une architecture basée sur les noms de données au lieu des adresses IP, elle est apparue en 2006 à l'Université de Californie, à Berkeley, et consiste à l'utilisation du routage par nom de données, comme le préconise TRIAD [section 3.2].

Les créateurs de l'architecture DONA se concentraient sur la résolution des problèmes des services réseau et de l'accès aux données, tout en tenant en compte des contraintes de la durabilité, la disponibilité et l'authenticité des données échangées. Cela a mené à la création d'un nouveau réseau dont les requêtes sont routées vers les nœuds les plus proches qui possèdent la donnée recherchée sans l'utilisation des serveurs DNS. Après avoir trouvé la donnée, une connexion TCP sera établie entre le demandeur et la source de données pour transférer les données, mais malheureusement, ce plan a également échoué.

### 3.3.3 L'architecture NetInf

Le projet **NetInf**[11] a été présenté pour la première fois en 2008 à Portland State University comme un nouveau réseau centré sur les données. Les principaux objectifs de ce projet étaient les suivants :

- La modélisation des données ;
- le dénomination des données ;
- et la recherche par nom de données plutôt que par le routage.

Pour réaliser ces objectifs, les concepteurs de NetInf utilisaient la méthode de la table de hachage distribuée (DHT) et un algorithme de recherche appelé Chimeras. EN effet, les données sont diffusées dans le réseau NetInf à l'aide du service de résolution de nom (NSR) qui est couplé d'un localisateur Net. Lorsqu'un consommateur demande une donnée quelconque, la requête sera classée par NSR et le reste du processus sera laissé au localisateur qui récupère les données.

### 3.3.4 L'architecture ICN

**ICN**[11] est une nouvelle architecture de réseau qui diffère radicalement de l'architecture IP, elle a été présentée en public pour la première fois en 2006 chez Xerox PARC par VAN Jacobson dans sa conférence de Google Tech Talk "A New Way to look at Networking". Les concepts principaux d'ICN visent à découpler la donnée recherchée de l'hôte

qui l'héberge en créant un système centrée sur les contenus plutôt que sur les adresses IP pour éliminer l'association des informations aux emplacements physiques.

Au cours de ces dernières années, une nouvelle conception d'architecture Internet, appelée Named Data Networking (NDN), a été proposée sous le cadre d'ICN, c'est l'une des cinq projets de recherche financés par la National Science Foundation des États-Unis.

### 3.4 Named Data Networking NDN

Named Data Networking (NDN) [1][13] est une nouvelle architecture internet orientée récepteur qui plonge ses racines dans le projet antérieur Content Centric Networking (CCN). Elle est issue des recherches empiriques sur les problèmes du réseau IP, ce qui la rend capable de s'adapter aux nouveaux modes de communication.

La forme de sablier qui a rendu la conception d'Internet d'origine robuste et puissante, représente aujourd'hui la brique de base de l'architecture NDN. En effet cette élégante forme a tout de même rendu NDN une conception forte et puissante et peut mener au futur proche à des stratégies évolutives claires et simples.

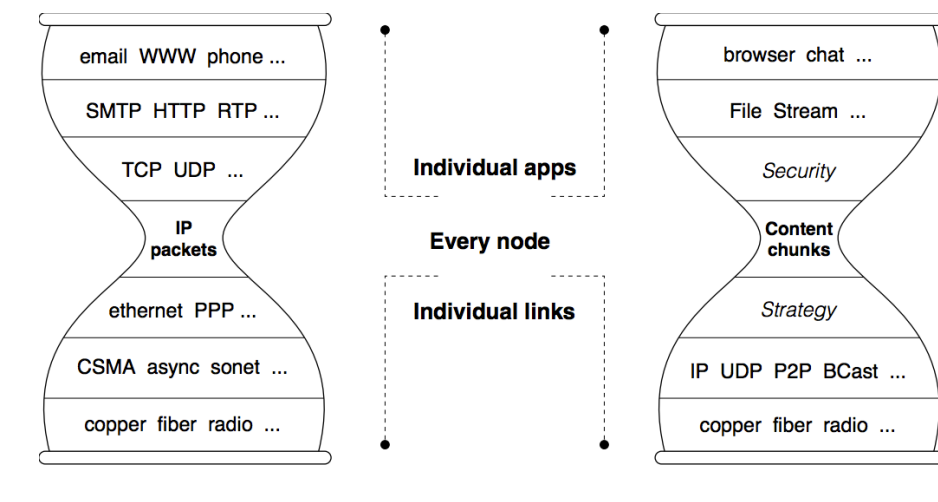


FIGURE 3.2 – L'architecture de sablier (internet et NDN). [1]

Comme il est montré dans la 3.2 , NDN est une superposition de couches universelles tout comme dans la conception IP ,c'est pourquoi que NDN peut être facilement intégrée dans n'importe quelle autre architecture, y compris IP, et inversement, n'importe quelle architecture peut être intégrée dans NDN, y compris IP. Cela veut dire que les services d'infrastructure IP qui ont pris des décennies d'évolution, telles que les conventions de dénomination DNS et l'administration des espaces de noms ou même les politiques et conventions de routage inter-domaine, peuvent être facilement utilisés par NDN.

Contrairement à l'ancienne architecture Internet, NDN n'a pas de couche de transport distincte. Elle a cependant transformé toutes les fonctions des protocoles de transport d'aujourd'hui en applications, ainsi que leurs bibliothèques et stratégies d'acheminement.

Par exemple les opérations de multiplexage et de démultiplexage entre les processus d'application se font directement à l'aide des noms de données au niveau de la couche NDN, tandis que l'intégrité et la fiabilité des données se vérifie directement par les processus d'application.

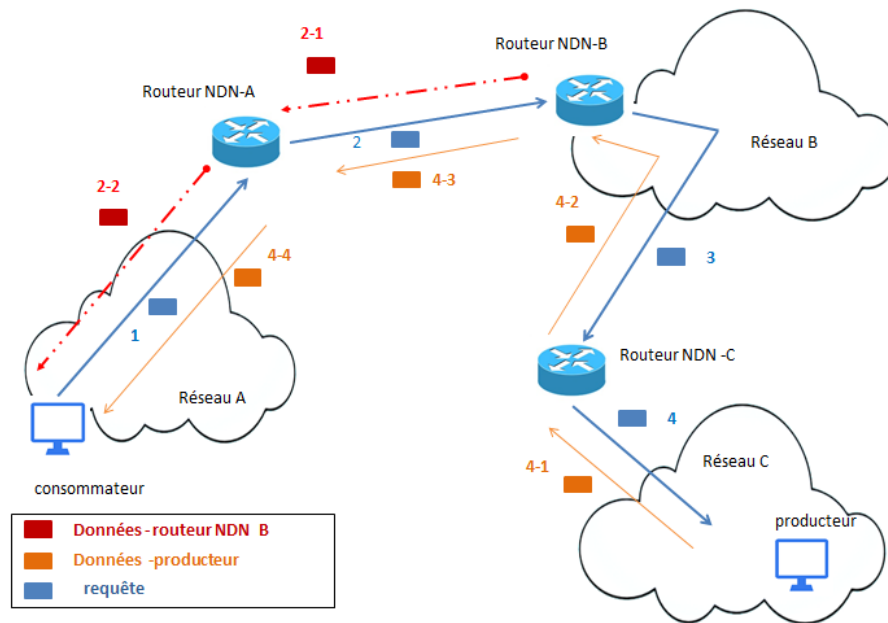


FIGURE 3.3 – Named data Networking NDN.[1]

Dans l'architecture NDN on aura plus besoin d'utiliser les serveurs DNS [figure 3.3], puisque les données sont entièrement indépendantes de leurs producteurs originaux, du fait que NDN offre la possibilité de mettre en cache toutes les données reçues à travers des requêtes antérieures. Comme il est montré dans la [figure 3.3], l'architecture NDN comprend trois principaux acteurs [1] :

- **Le producteur ( publisher )** : c'est le nœud source de données ;
- **facilitateurs ( facilitator )** : des nœuds intermédiaires (routeurs) chargés de recueillir, de sauvegarder et de relayer les données générées par les producteurs ;
- **le consommateur** : le nœud qui envoie la requête aux producteurs/ facilitateurs.

## 3.5 Les noms de données

Les noms de données représentent la pièce la plus importante dans la conception NDN [1][13], Ceci est dû au fait que NDN tente d'élargir l'écart entre l'identificateur de données (le nom) et le localisateur (le moyen de récupération). Dans les réseaux IP avec DNS, la différence est marginale, une URL est à la fois un identifiant et un localisateur de données. NDN insiste plutôt sur le fait qu'un morceau de données ne devrait avoir qu'un identificateur unique, même s'il est disponible dans plusieurs emplacements. En outre, les noms sont opaques au réseau NDN, c'est-à-dire, les routeurs ne comprennent pas la signification des noms. Cela permet aux applications de définir leurs propres systèmes de nommage indépendants du réseau.

Par exemple le fichier de ce mémoire peut être nommé de la façon suivante [Figure 3.4] : **Dz/lagh-univ/fac-science/dprt-info/mémoire/VNDN** Où le caractère '/' ne représente qu'un délimiteur entre les composants et ne fait pas partie du nom. Cette structure hiérarchique présente deux avantages, premièrement, elle permet aux applications de représenter la relation entre les éléments de la donnée, Par exemple, le chapitre 2 de ce mémoire peut être nommé **Dz/lagh-univ/fac-science/dprt-info/mémoire/VNDN/2**

Deuxièmement, elle permet l'agrégation des noms qui est essentielle dans l'échelle du système de routage, supposant par exemple que NDN considère l'université de Laghouat comme un système autonome, on peut alors distribuer le préfixe de nom **Dz/lagh-univ/** via les protocoles de routage d'une manière semblable à la distribution de préfixes IP dans Internet d'aujourd'hui. Puisque les noms NDN sont structurés de manière hiérarchique, il faut utiliser le mécanisme «longest prefix matching» pour faire le routage. Si un nœud NDN demande par exemple **Dz/lagh-univ/fac-science/dprt-info/mémoire/VNDN/**, un nœud qui a la donnée **Dz/lagh-univ/fac-science/dprt-info/mémoire/VNDN/2/1-3/** dans son cache peut satisfaire la requête, donc les routeurs n'ont pas besoin de stocker deux entrées distinctes pour **/VNDN/** et **/VNDN/2/1-3/**, car les requêtes des deux préfixes de nom seront envoyées vers la même interface.

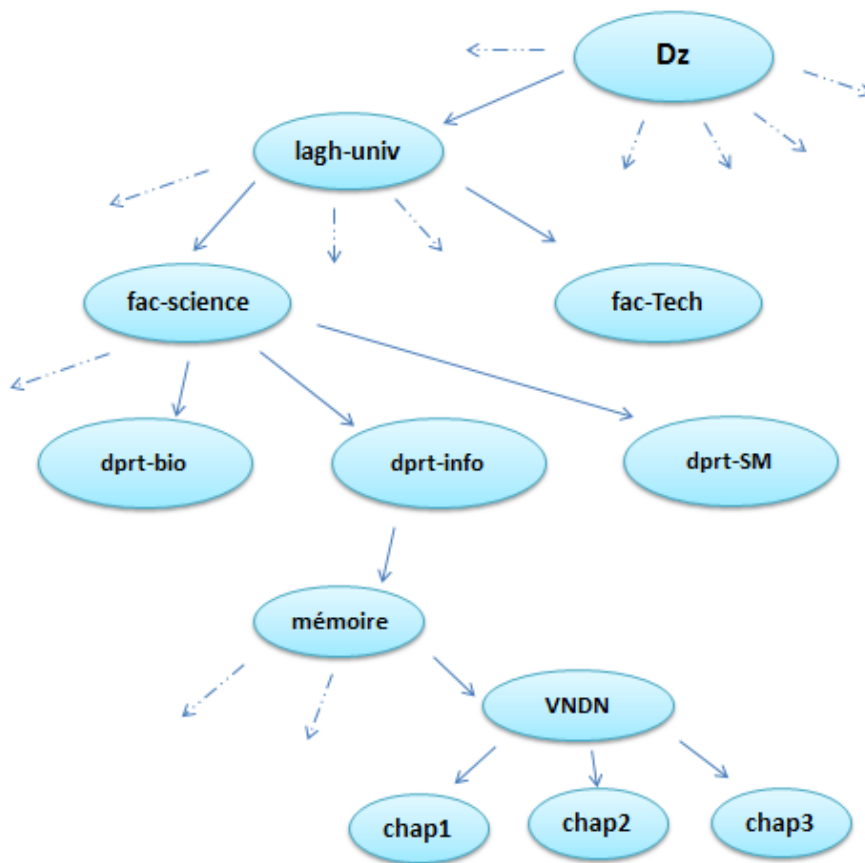


FIGURE 3.4 – Les noms hiérarchiques.

### 3.6 Les paquets NDN

Dans l'architecture NDN, il existe deux types distincts de paquets :

- Paquet de requête (interest packet – IntPK)[1] ;
- paquet de données (data packet - DataPK)[1].

Les structures des deux paquets NDN, sont montrées dans la figure 3.5 ci-dessous :

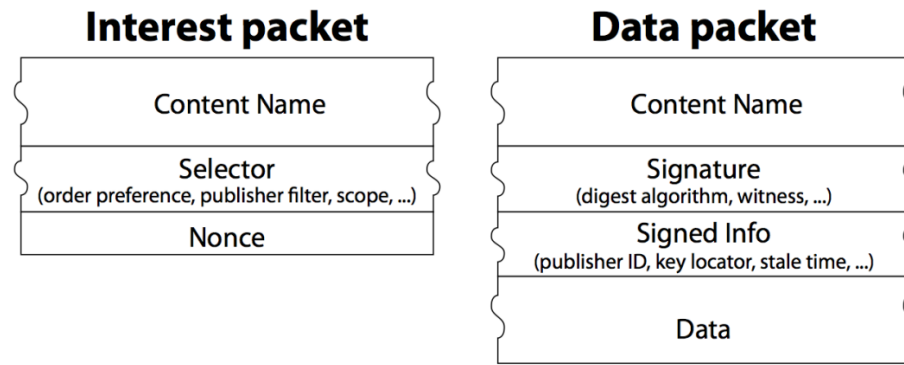


FIGURE 3.5 – La structure des paquets.[1]

### 3.6.1 Paquet de requête - interest packet (IntPK)

Ce paquet est composé de 3 parties :

Champs	Description
Content Name	contient un identifiant de l'objet désiré (un nom). remplace le champ Identification dans le datagramme IP qui est actuellement utilisé.
Selectors	C'est un champ optionnel qui indique : l'ordre des paquets, préférences, etc.
Nonce	C'est un nombre aléatoire attribué au niveau de la PENDING INTEREST TABLE (PIT). Pour assurer l'unicité du paquet et éviter le problème de boucle.

Le consommateur met le nom de la donnée qu'il recherchait dans un IntPK et l'envoie au réseau. Les routeurs utilisent ce nom pour transmettre la requête vers les producteurs.

### 3.6.2 Paquet de données – data packet (DataPK)

Le paquet de données se compose de 4 différentes parties :

Champs	Description
Content Name	contient le même identifiant que celui de IntPK
Signature	la signature de la clé du producteur qui lie le nom et la donnée. pour assurer la cohérence des données.
Signed info	représente la clé de l'éditeur qui est exigée par le nœud récepteur.
DATA	ce champ contient la donnée demandée.

Une fois le IntPK atteint un nœud qui a les données demandées, ce dernier va répondre par un DataPK qui suivra en sens inverse le chemin emprunté par le IntPK pour arriver au consommateur.

## 3.7 Architecture des routeurs NDN

Les routeurs NDN maintiennent trois principales structures de données [1] :

- Content store - CS;
- Pending Interest Table - PIT;
- Forwarding Information Base – FIB.

### 3.7.1 Content store - CS

Pour satisfaire les futures requêtes, les routeurs NDN conservent toutes les informations qui passent par eux dans une structure de données appelée Content stores (CS) [figure 3.6]. Cela permet une meilleure utilisation de la bande passante et réduit le délai de bout en bout. Généralement le CS utilise deux types d'algorithmes de gestion de stockage, LFU (Least Frequently Used), qui supprime les données qui ont la plus petite fréquence d'utilisation, et l'algorithme LRU (Least Recently Used), qui purge les paquets de données qui n'ont pas été utilisées depuis longtemps.

### 3.7.2 Pending Interest Table - PIT

La PIT (Pending Interest Table) [figure 3.6], sauvegarde tous les IntPK qui sont transmis vers des éventuelles sources de données, pour pouvoir ensuite renvoyer les DataPK vers les consommateurs. Une fois que le DataPK soit transmis, on efface l'entrée PIT qui correspond à celui-ci. Les entrées PIT seront aussi supprimées, après l'expiration d'un délai prédéterminé.

### 3.7.3 Forwarding Information Base - FIB

Les FIB (Forwarding Information Base) [figure 3.6], dans les routeurs NDN sont utilisés pour acheminer les requêtes vers les producteurs de données. Ils contiennent des protocoles de routage basés sur les noms de données afin de déterminer les interface vers lesquelles les paquets seront envoyés .

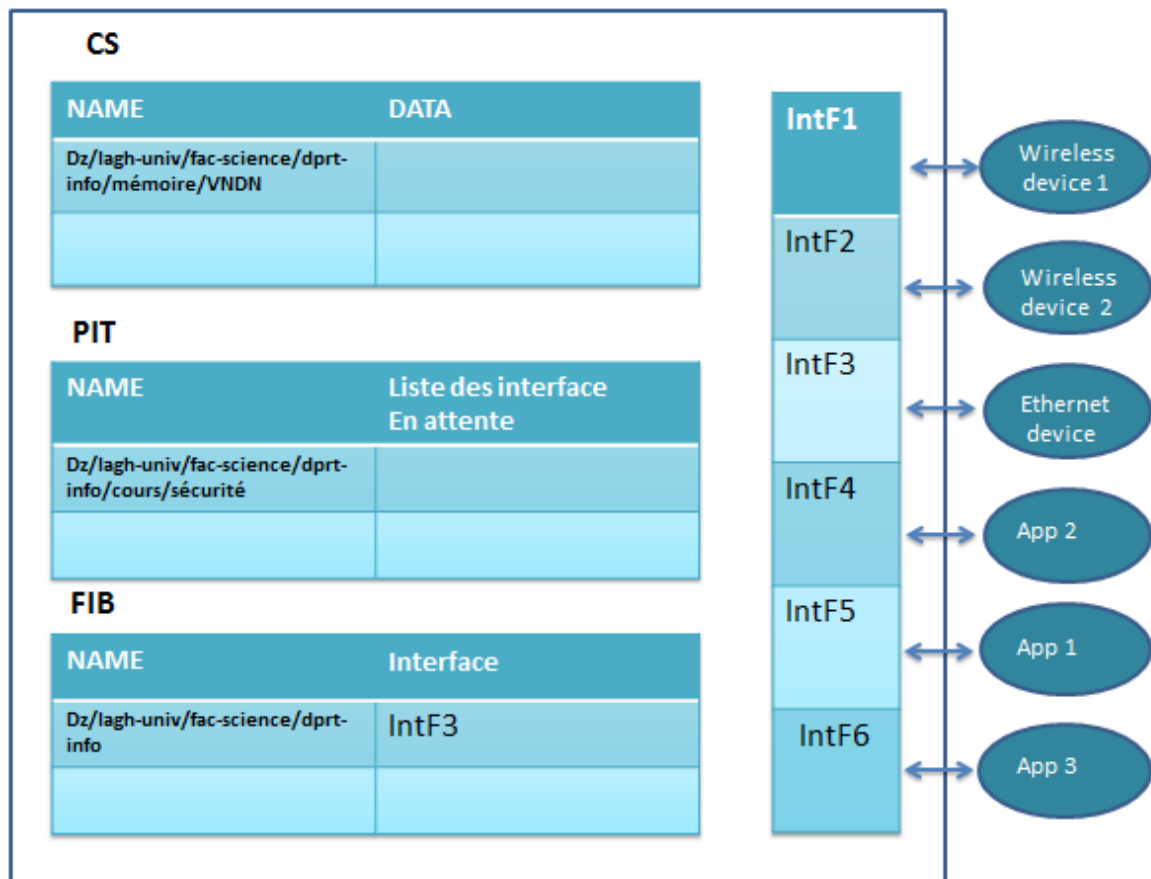


FIGURE 3.6 – L'architecture d'un routeur NDN. [1]

### 3.8 Le principe de fonctionnement NDN

Comme mentionné dans la section précédente, le routage dans les réseaux NDN est basé sur les noms de données et non pas sur les adresses IP. Et vu que les espaces de noms sont illimités, le problème d'épuisement de l'espace d'adressage qui est dû à l'accroissement du nombre d'utilisateurs et des serveurs Internet sera définitivement éliminé. Ce nouveau système de routage élimine aussi le problème de NAT traversal dès lors qu'on aura plus besoin d'utiliser les adresses IP pour la récupération ou l'envoi de données. En outre la mobilité des nœuds ne perturbera plus la communication puisque les noms restent toujours les mêmes. Ainsi, Il est à noter que tous les algorithmes de routage traditionnels qui fonctionnent bien dans le réseau IP, tels que Link-state, Path-Vector ou même OSPF peuvent être facilement adoptés par le réseau NDN.

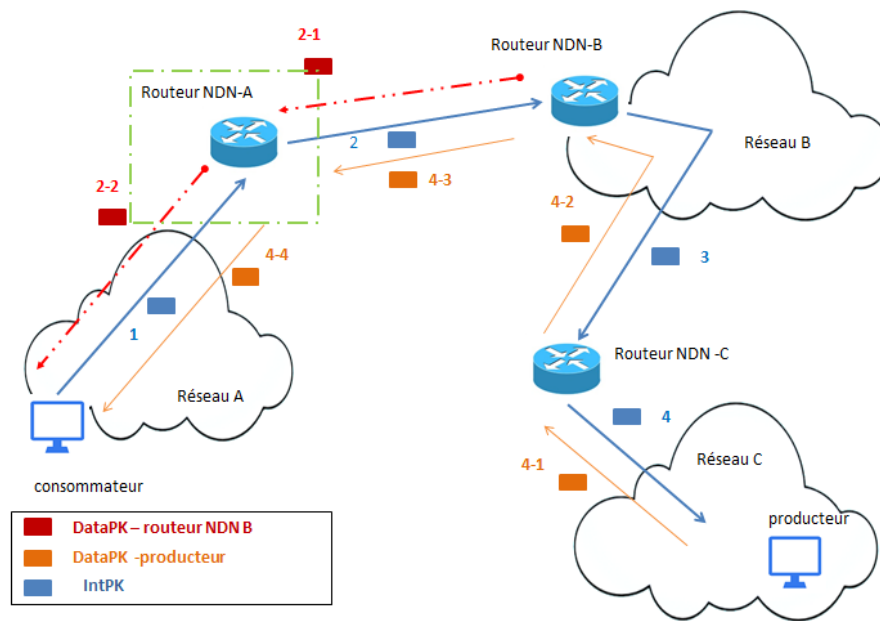


FIGURE 3.7 – Le fonctionnement de NDN.

Le processus de récupération de données [11][12][voir figure 3.7] dans les réseaux NDN se fait en utilisant les deux paquets : Interest packet (IntPK) et Data packet (DataPK). Il commence toujours par un nœud consommateur qui diffuse un IntPK généré avec un certain préfixe de nom. Le traitement ultérieur des paquets IntPK et des DataPK au niveau de chaque nœud NDN dans le réseau est effectué à l'aide de trois structures de données mentionnées dans la section 3.3. Soit un nœud NDN :

- À la réception d'un IntPK par , ce dernier va tout d'abord rechercher dans son CS le plus long préfixe de nom qui correspond à celui spécifié dans le paquet reçu ;
- si c'est le cas, N répond directement par un DataPK contenant la donnée demandée ;
- sinon, la PIT sera consultée, s'il y a déjà quelqu'un d'autre qui a demandé cette données, c'est-à-dire qu'il existe déjà une entrée correspondante dans la PIT, va ajouter le IntPK à la liste des interfaces requérantes dans la PIT, ensuite il le rejette ;
- si aucune correspondance n'est trouvée, une nouvelle entrée PIT est créée, puis la FIB sera consultée pour déterminer vers quelle interface on va transmettre le paquet ;
- la recherche dans la FIB sera effectuée de la même manière que dans le CS ou PIT, c'est-à-dire : recherche par longest prefix matching ;
- si un routeur se trouve incapable de satisfaire un Interest Packet, par exemple, dans le cas où il n'y a pas d'entrée d'acheminement dans la FIB, ou dans le cas d'une congestion dans le réseau, le routeur va rejeter le paquet en envoyant un accusé négatif (NACK) à son voisin qui lui a transmis le paquet, pour que ce dernier explore d'autres chemins alternatifs.

La [figure 3.8] ci-dessous montre le processus exécuté par le routeur NDN-A après de la réception d'un IntPK :

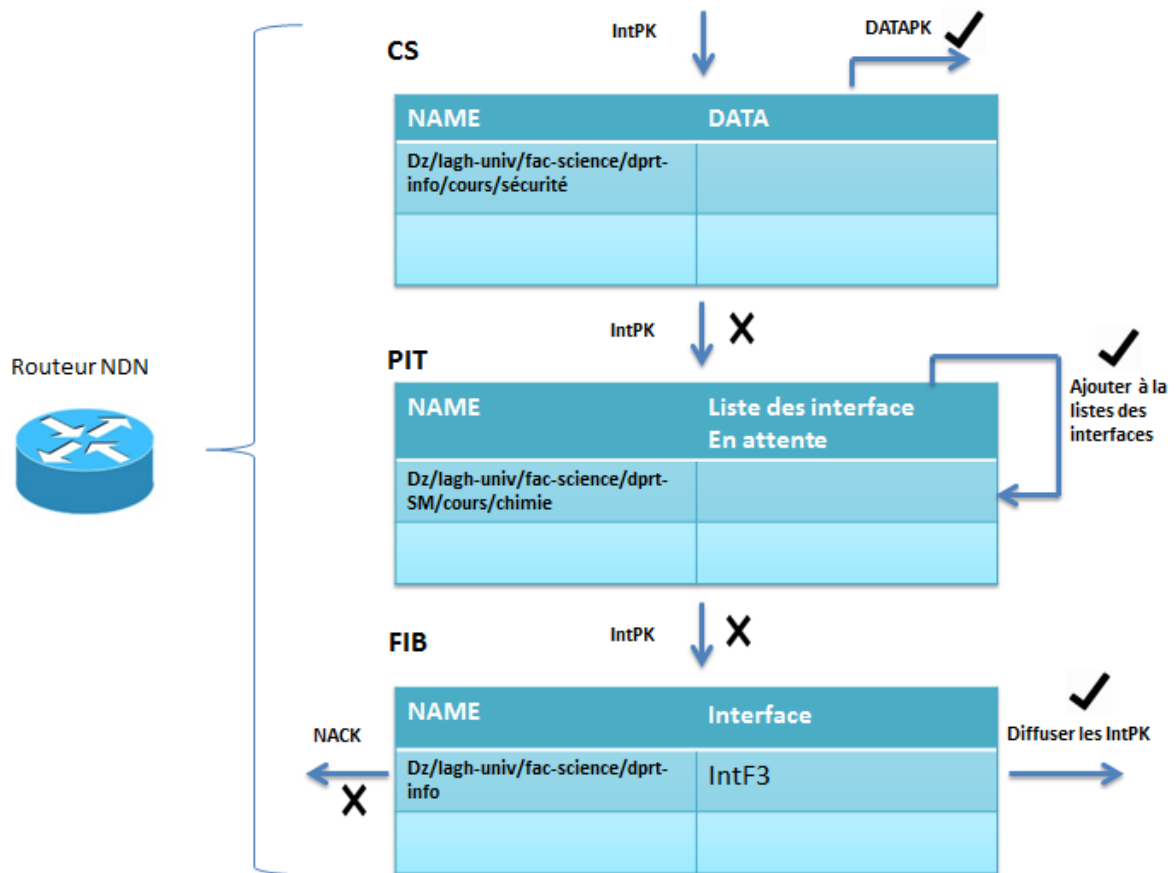


FIGURE 3.8 – Le processus exécuté par un routeur NDN lors de la réception d'un IntPK.[13]

Lorsque le DataPK atteint une interface d'un nœud [voir figure 3.9] :

- Le nœud va consulter sa PIT pour vérifier l'existence d'une entrée correspondante ;
- si c'est le cas, il sauvegarde la donnée dans son CS afin de servir les futures requêtes, et transfère le paquet vers les interfaces qui sont listés dans la file d'attente de la PIT ensuite il la purge ;
- sinon il ignore tout simplement le DataPK.

Le mécanisme implémenté dans la PIT permet aux routeurs de surveiller le processus de récupération des paquets à partir des différentes interfaces et de détecter les boucles de routage. Un nombre aléatoire appelé nonce sera attribué à chaque paquet de requête reçu, et en le comparant avec celui du DataPK, on peut facilement conclure à l'existence du problème de boucle et de l'éliminer directement. Cela permet d'utiliser librement plusieurs chemins vers le même producteur de données. Sans avoir besoin de mettre en œuvre des

protocoles connexes au niveau des nœuds pour résoudre ce problème, comme dans le cas du réseau IP.

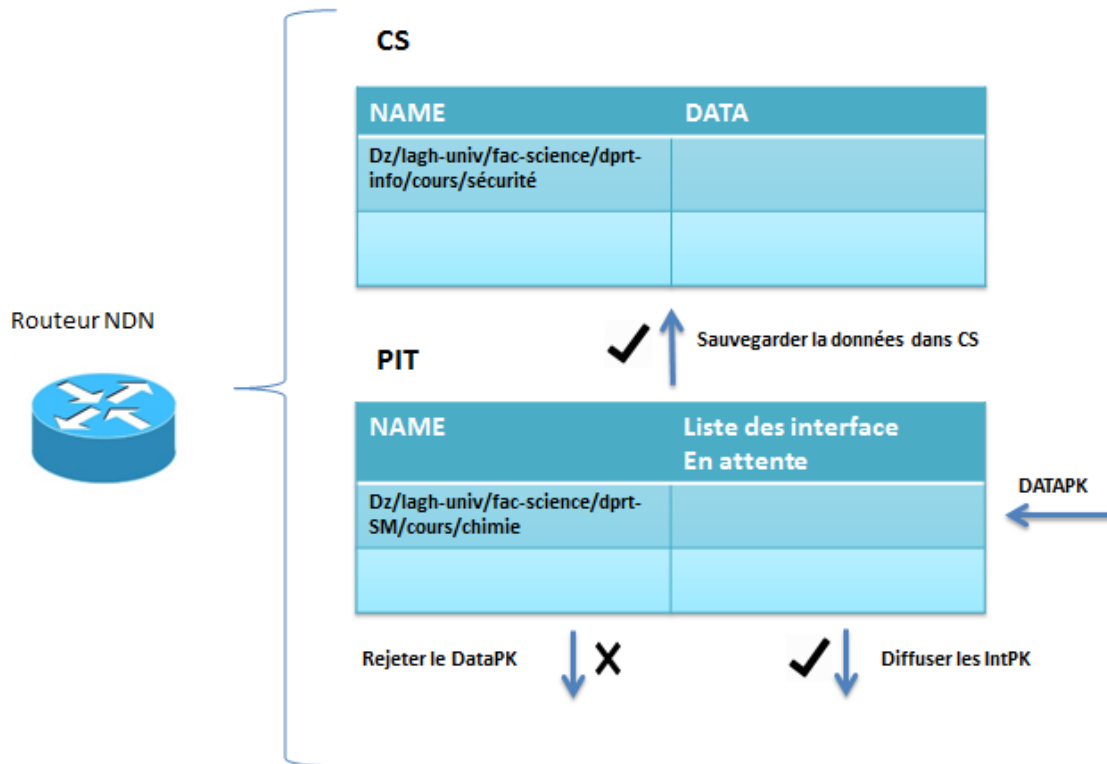


FIGURE 3.9 – Le processus exécuté par un routeur NDN lors de la réception d'un DataPK.[13]

### 3.9 Les nouveaux réseaux NDN

Récemment, il y a eu une nouvelle tendance émergente dans le monde de télécommunication qui consiste à l'extension d'Internet à des objets et à des lieux du monde physique, c'est ce qu'on appelle internet of Things (IoT) ou Internet des objets. Selon la définition de l'Union internationale des télécommunications, Internet of Things (IoT), est une «infrastructure mondiale pour la société de l'information, qui permet de disposer de services évolués en interconnectant des objets (physiques ou virtuels) grâce aux technologies de l'information et de la communication interopérables existantes ou en évolution».

En effet, il semble que la réalisation de cette nouvelle vision d'internet en utilisant l'ancien paradigme de communication qui repose sur le protocole IP[14], est difficile surtout dans le contexte des dispositifs fortement hétérogènes et de leur connexion intermittente aux autres réseaux. Ce problème est exacerbé par l'obligation de l'utilisation de plusieurs interfaces de communication et une multitude de régimes de sécurité.

Si on prend l'exemple des maisons intelligentes, les technologies IoT ont été largement intégrées pour surveiller et contrôler l'environnement des maisons individuelles (les thermostats intelligents, les détecteurs de fumée, les caméras de sécurité ; Switchs de lumière sans fil, etc). Dans de nombreux cas, ces technologies permettent la surveillance et le contrôle à distance à l'aide de smartphones ou d'ordinateurs de n'importe où dans le monde. Les nouvelles applications de santé prennent aussi une large partie dans le domaine de développement des IoT, ils utilisent de très petits systèmes intégrés dans des chips qui peuvent être adaptés à des appareils portables (par exemple : les bracelets et les montres intelligents, etc). De tels dispositifs de santé personnels intègrent plusieurs capteurs pour surveiller l'activité physique, la température corporelle et la fréquence cardiaque d'un utilisateur, qui sont signalés au téléphone intelligent de l'utilisateur via Bluetooth LE ou via des technologies sans fil similaires. Dans de nombreux cas, les données personnelles sont téléchargées et sauvegardées dans des entrepôts de données hébergés par des Cloud pour faire des analyses plus approfondies par des logiciels ou des médecins.

Le paradigme NDN répond naturellement aux exigences inhérentes des applications IoT[14], car il fournit un protocole de réseau unique qui peut faciliter le déploiement d'Internet of Things, et cela en fournissant une sémantique de requête-réponse fondée sur les noms de données au lieu des adresses IP, ce qui permet aux applications de désigner clairement et facilement les objets du réseau , ainsi que de transférer les paquets de données de ces objets directement en fonction de leurs noms. Les noms NDN sont structurés de façon hiérarchique et flexible, de façon à permettre au nœud de traiter simplement chaque composant du nom de la données en tant qu'une suite de bits. La longueur des noms est également flexible, de sorte que les applications peuvent utiliser des noms courts pour réduire les overheads.

La structure hiérarchique des noms peut être exploitée pour assurer aussi la sécurité des réseaux IoT [14] en utilisant un mécanisme conçu de façon flexible pour mieux gérer la confiance des utilisateurs. Et cela en intégrant la sécurité dans la donnée elle-même plutôt que de lutter pour définir et gérer la sécurité dans les sous-réseaux, et les canaux de communication. La solution de sécurité que propose NDN pour les réseaux IoT constituent une alternative robuste à la création d'authentification granulaire au niveau des paquets et au contrôle d'accès.

L'agrégation des données devient une tâche aussi facile avec le paradigme NDN [14] qu'avec l'ancien paradigme TCP/IP. Tout d'abord les applications IoT définissent les conventions de dénomination et d'encodage des données traitées par le capteur (i.e le type de données, l'emplacement de mesure, et le moment où elles sont prises,etc). Ensuite Les passerelles peuvent extraire périodiquement les données du capteur.

## 3.10 Conclusion

NDN représente une nouvelle architecture internet, qui est bien compatible avec l'architecture IP actuellement utilisée. Elle suit une stratégie évolutive claire et simple, afin de fournir dans un futur proche une nouvelle gamme de services et d'avantages tels que la mise en cache, la simplicité de configuration des périphériques réseau et la sécurité des données. Cela qui permettra aux internautes de surfer en toute confidentialité en réduisant ainsi le problème de congestion et améliorant la vitesse de livraison.

L'architecture NDN peut être étendue au réseau VANET afin de résoudre les problèmes qui sont difficiles à traiter dans l'architecture traditionnelle IP, et de contribuer ainsi à l'amélioration des performances des systèmes de transport, la sécurité et le confort des usagers. Dans le chapitre suivant, nous allons introduire l'architecture Vehicular Named Data Networking (VNDN) et ses différentes stratégies d'acheminement de données.

## Chapitre 4

# Vehicular Named Data Networking VNDN

### 4.1 Introduction

Dans les VANET traditionnels, chaque véhicule dans le réseau doit se voir attribué un identifiant (ID) unique (i.e. adresse IP). Le véhicule source de données utilise ces IDs uniques pour localiser précisément les véhicules de destination pour leur transmettre les informations. Il est en effet nécessaire que les deux bouts de communication, source et destination, établissent et sécurisent le canal avant d'acheminer les informations sensibles entre eux. Cette tâche devient très difficile en présence de la mobilité des nœuds car il est impossible de conserver le même ID d'un hôte ainsi que de maintenir le(s) chemin(s) de communication lorsque les nœuds se déplacent d'une zone à une autre.

Dans le chapitre précédent, on a défini NDN comme étant une nouvelle architecture internet orientée récepteur, dont les communications sont centrées sur les noms de données plutôt que sur les adresses IP. Dans le présent chapitre, on va présenter et discuter les différents designs VNDN (Vehicular Named Data Networking) proposées par différents groupes de travail.

### 4.2 Apports de NDN aux réseaux VANET

En raison de ces fortes potentialités cette architecture a été récemment étendue aux réseaux VANET par de nombreux groupes de travail pour pouvoir surmonter les défis que peut poser un réseau véhiculaire, tels que la connectivité intermittente qui est due à la mobilité des nœuds. En effet, NDN peut fournir plusieurs avantages aux réseaux VANET et représente un substituant valide à l'ancien paradigme d'Internet centré sur les hôtes, qui est mal adapté à ce type de réseau fortement dynamique. Parmi ces avantages on peut citer [15]

- **Configuration réseau**

Le paradigme NDN permet aux nœuds du réseau de communiquer entre eux sans aucun besoin de configuration à priori des paramètres réseau (i.e. : adresse IP, serveur de noms, etc). Cela est particulièrement utile dans les environnements dynamiques où la configuration statique n'est plus possible en raison de la mobilité des nœuds, et des solutions comme l'IP mobile ne sont pas satisfaisantes.

- **Diversité des applications**

NDN est particulièrement approprié pour l'ensemble des applications pertinentes aux positions géographiques. En effet, ce type d'applications est propre aux environnements véhiculaires dont les véhicules peuvent être impliqués dans un processus de communication en fonction de leurs positions dans une zone donnée indépendamment de leurs identités ou de leurs adresses IP comme par exemple : les applications du trafic routier, ou de la météorologie, ou même de la publicité sur les points d'intérêts proches, etc.

- **Conservation facile et efficace des données dans le réseau**

Les routeurs NDN sont capables de conserver toute donnée qui passe par leurs caches pour satisfaire les futures requêtes. Cette nouvelle technique peut être appliquée directement sans coût supplémentaire grâce aux fonctionnalités puissantes des dispositifs véhiculaires, non limités par l'énergie et les capacités de stockage.

- **Échange asynchrone de données**

L'architecture NDN supporte nativement l'échange asynchrone de données entre les nœuds finaux du réseau, puisqu'elle est conçue de façon à permettre aux nœuds mobiles de servir de relais entre les zones déconnectées, et cela en utilisant les données conservées dans leurs mémoires cache. Cette caractéristique permet d'élargir les zones de communication, et l'échange de données même sous une connectivité intermittente ce qui augmente la probabilité de trouver des réponses aux requêtes qui se propagent dans le réseau.

- **Mise en œuvre facile**

Les véhicules, les RSU et les serveurs locaux peuvent être équipés facilement d'une pile de type NDN sans aucune modification des Infrastructures du réseau, ce qui réduit fortement le temps de mise sur le marché de ce paradigme prometteur.

On appelle l'architecture résultante de la fusion des VANET et NDN : Vehicular Named Data Networking (VNDN). Il est à noter que toutes les opérations NDN restent valables dans cette nouvelle architecture hybride, et donc le mécanisme de transmission de données vu précédemment va rester toujours valable. Néanmoins, de nombreux groupes de recherche ont investi particulièrement dans le développement de ce mécanisme que chacun a proposé son propre modèle VNDN. Dans la suite de ce chapitre, nous allons présenter 4 designs VNDN développés par différents groupe de travail.

### 4.3 Le 1<sup>er</sup> design VNDN Vehicular Named Data Networking

Les réseaux véhiculaire VANET possèdent deux caractéristiques fondamentales : la connectivité intermittente ad hoc et la capacité de transporter physiquement les données. Les auteurs ont développé un design VNDN [16] tout en gardant ces deux principales caractéristiques, et en l'employant de façon à permettre aux véhicules de jouer à la fois quatre différents rôles [voir figure 4.1] :

- Consumer : consommateur de données ;
- producer (publisher) : producteur de données ;
- forwarder : Lorsqu'il est connecté aux infrastructures routières ou à d'autres véhicules ;
- data-mule : mule de données, lorsqu'il transporte des données d'une zone à une autre en n'ayant aucune connectivité à quelqu'un d'autre.

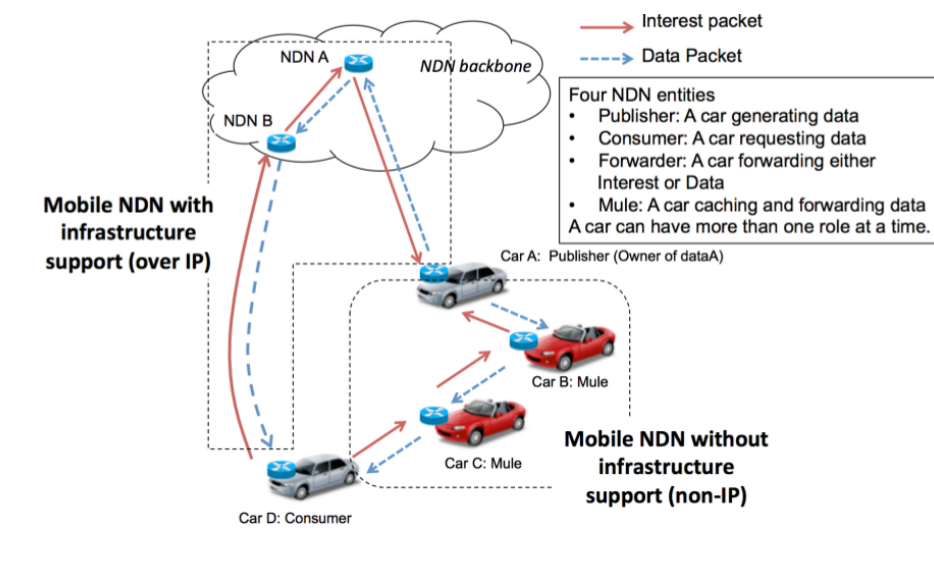


FIGURE 4.1 – Architecture VNDN [17].

En supprimant les contraintes de la pile protocolaire TCP/IP, le paradigme NDN pourrait être très bénéfique aux réseaux VANET. Cependant pour adapter les opérations NDN à l'environnement véhiculaire plusieurs modifications ont été apportées par le groupe de travail :

- Tout d'abord, étant donné que toutes les communications VANET se font sur des canaux sans fil en mode Broadcast (diffusion), et qu'un véhicule se dispose d'une grande capacité de stockage. Dans ce cas, le véhicule aura la possibilité de conserver toute donnée reçue indépendamment du fait qu'il ait une entrée PIT correspondante à cette donnée ou s'il en a besoin pour lui-même ;

- deuxièmement, les DataPK peuvent être transportés par les véhicules Data-mule même lorsqu'ils n'ont pas de connectivité avec le réseau. Donc, les données peuvent se propager dans le réseau et s'éloigner des emplacements de leurs producteurs, soit à travers des requêtes (IntPK), ou à travers les mouvements des véhicules ;
- enfin, l'environnement dynamique des réseaux VANET et la connectivité intermittente entre les véhicules en mouvement rend difficile, voire impossible, d'exécuter des protocoles de routage pour construire et maintenir des entrées stables dans la FIB. Par conséquent, VNDN doit développer d'autres moyens pour guider le transfert des IntPK.

### 4.3.1 Architecture VNDN

Dans cette section, nous allons détailler les différentes composantes de ce design VNDN [voir figure 4.2] ainsi que leurs fonctionnalités :

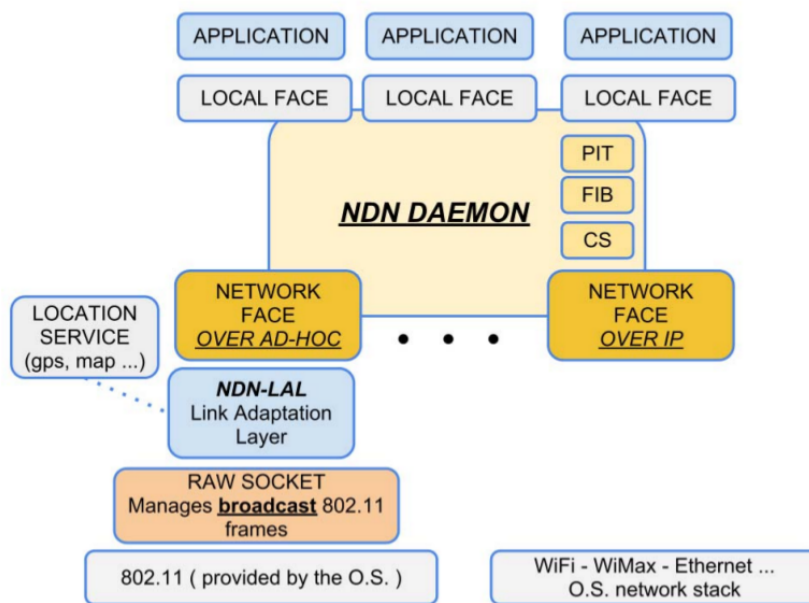


FIGURE 4.2 – Implémentation VNDN par[16].

#### 4.3.1.1 NDN Daemon

Il fournit les fonctionnalités de base de NDN en maintenant les trois structures de données CS, PIT et FIB[16] :

- CS : sauvegarde toutes les données qui passent par le véhicule, au lieu de ne sauvegarder que les données qui correspondent aux entrées PIT ;
- PIT : fournit les mêmes fonctionnalités qu'une PIT classique ;

- FIB : l'implémentation actuelle prend une simple approche si le véhicule est équipé de plusieurs interfaces. Cette approche consiste à diffuser les IntPK vers toutes les interfaces qui sont disponibles à l'instant  $t$ , donc la FIB n'a pas les mêmes fonctionnalités qu'une FIB classique, autrement dit elle n'est pas utilisée dans le routage ou dans la décision d'envoi des IntPK vers les éventuels producteurs de données ;
- la technique de recherche dans NDN daemon est similaire à celle de l'NDN classique (Prefix matching).

#### 4.3.1.2 NDN Local Faces

Il s'agit des interfaces qui se situent entre la couche application et le NDN Daemon[16]. Ils prennent en charge l'enregistrement des applications auprès de la FIB, autrement dit, la FIB crée des entrées contenant des prefix produit par les applications et les faces pour accéder à ces prefix. Les Local Faces supporte aussi la distribution des data et Interest packet.

#### 4.3.1.3 Network Faces

Ils fournissent Il supporte plusieurs technologies de communication[16]. Ainsi, il utilise IEEE 802.11 en mode ad hoc (IBSS) pour les communications V2V, et WiMAX, 3G et Wi-Fi pour les communications V2I. Pour la connectivité 3G, le NDN Network Face fournit l'adaptation nécessaire à l'IP tunneling entre le mobile et certains nœuds NDN du réseau.

#### 4.3.1.4 Link Adaptation Layer (LAL)

Elle est considérée comme la couche 2.5, conçue pour tirer efficacement les avantages et les potentialités des mécanismes de la couche 2[16]. La couche LAL envoie tous les paquets en mode diffusion, en utilisant la norme 802.11 pour transporter directement les paquets NDN.

#### 4.3.1.5 Location service

Fournit des fonctionnalités de géocodage inverse, ainsi que d'autres fonctions de calcul de distance et de position au LAL, qui les utilise dans les communications véhiculaires et pour prendre des décisions sur le transfert des paquets[16]. Par ailleurs, certaines applications peuvent choisir de coder les informations de localisation dans les noms de données lorsque le contenu concerne une zone limitée, comme par exemple les informations du trafic ou de stationnement.

### 4.3.2 Amélioration de la diffusion Wi-Fi pour les communications V2V

On utilise la diffusion Wi-Fi de la couche 2 pour toutes les communications V2V. Cependant, la norme IEEE 802.11 actuelle ne peut fournir ni la prévention de collision ni un mécanisme de détection ou de résolution de ce problème. Par ailleurs, la diffusion Wi-Fi peut parfois subir de fortes pertes de données, dues à la nature dynamique des réseaux véhiculaires, cette perte peut être exacerbée par les changements rapides de la topologie du réseau et les durées courtes des liens de communication. Pour assurer une diffusion efficace et robuste au problème de perte, Ils ont développé un ensemble de mécanismes pour fournir un support de diffusion Wi-Fi dans les communications VANET. Ce support de diffusion Wi-Fi est couplé par un algorithme de transfert de paquets, qu'on va le décrire ci-après. Étant donné que chaque véhicule est équipé d'un GPS et d'une carte numérique, il utilise une simple stratégie de greedy forwarding pour diffuser les paquets IntPK dans toutes les directions de la manière suivante :

- Chaque paquet IntPK, noté  $i$ , porte les informations de localisation de son expéditeur  $S$  ;
- Lorsque  $i$  est reçu par plusieurs véhicules voisins, le véhicule le plus éloigné de  $S$  le rediffuse ; Les autres récepteurs ne font rien ;
- En outre,  $S$  doit savoir si le paquet  $i$  a été reçu et transmis, sinon  $S$  doit retransmettre  $i$ .

Les auteurs ont implémenté les mécanismes ci-dessous (voir figure 4.3) dans la couche Link Adaptation Layer (LAL) :

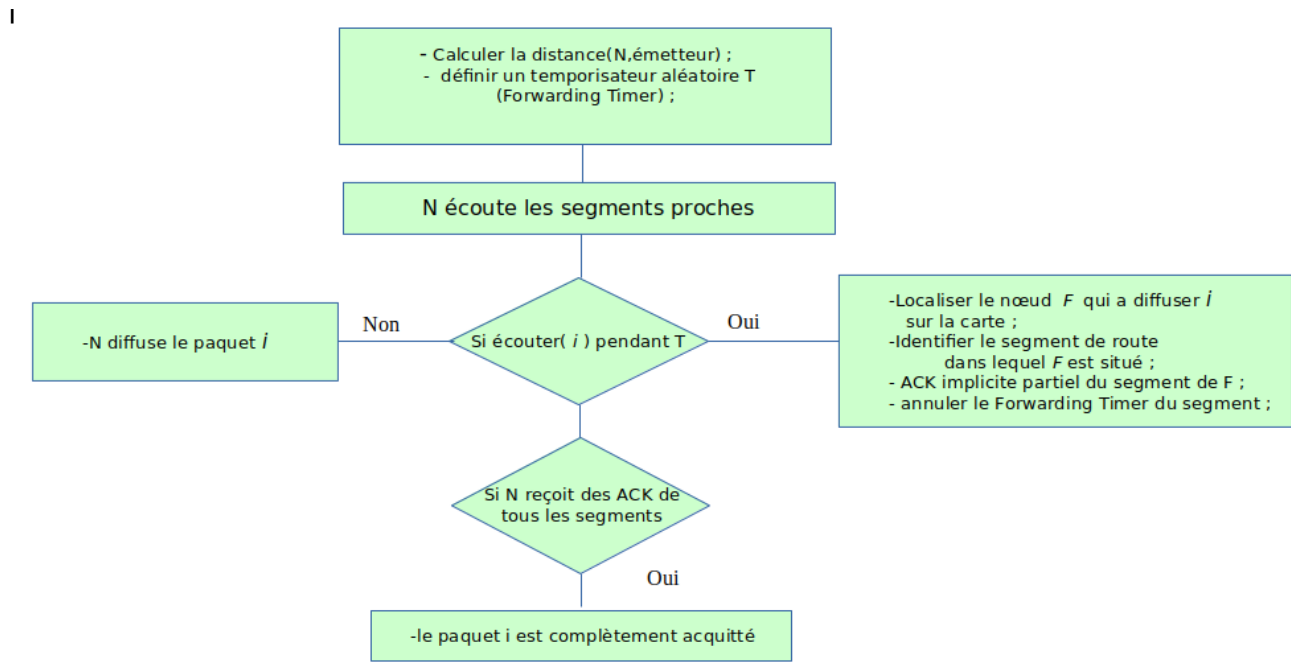


FIGURE 4.3 – Schéma illustratif du processus de diffusion d'un IntPK [16].

- La couche LAL calcule la distance entre l'émetteur et lui-même, en utilisant le Location service GPS, puis définit un temporisateur aléatoire (Forwarding Timer) en fonction de cette distance de telle sorte que plus la distance est courte, plus l'attente est longue.

La couche LAL combine deux composantes différentes pour calculer le Forwarding Timer :

Une composante déterministe :

1

D (émetteur, récepteur)

Où la distance D est calculée en utilisant le service de localisation. Et une petite composante aléatoire utilisée pour randomiser le temps de transmission.

La première composante favorise les voitures plus éloignées du dernier saut, le second réduit les chances de collision entre les nœuds à la même distance. Pour limiter la diffusion excessive des IntPK sur le réseau véhiculaire, on a ajouté un compteur de sauts dans l'en-tête LAL qui est incrémenté à chaque saut. Un IntPK est abandonné lorsque son nombre de sauts dépasse une limite prédéfinie n

Le mécanisme d'attente aléatoire pondéré (Forwarding Timer) permet au nœud le plus éloigné (dans la [figure 4.4] c'est le véhicule rouge) de l'émetteur S d'acheminer le paquet  $i$ , ce qui entraîne une propagation rapide des paquets.



FIGURE 4.4 – Processus de diffusion d'un IntPK. [16]

- Pendant ce temps d'attente, si  $N$  entend le transfert de  $i$  par un autre nœud  $F$ , il utilise un géocodage inverse pour localiser  $F$  sur la carte et identifier le segment de route dans lequel  $F$  est situé ;
- Cette transmission est considérée par  $N$  comme un acquittement partiel implicite de  $i$  ;
- si  $N$  reçoit des accusés de réception partiels implicites de chacun des segments de route qui sont proches de l'endroit où  $N$  est situé (à l'exception de celui d'où le paquet a été initialement diffusé, c'est-à-dire le segment de  $S$ ), il considère le paquet  $i$  comme complètement acquitté et annule son Forwarding Timer ;
- sinon, lorsque le temporisateur expire,  $N$  diffuse le paquet  $i$  ;
- de même, le paquet transmis peut être entendu aussi par  $S$ , et si  $S$  n'entend pas des acquittements partiels implicites de tous les segments de route proches de son emplacement, il retransmet le paquet ;
- les tentatives de retransmission sont bornées par une limite prédéfinie  $n$ , le paquet sera abandonné après  $n$  tentatives infructueuses.

La première composante favorise les voitures plus éloignées du dernier saut, la seconde réduit les chances de collision entre les nœuds à la même distance.

Pour limiter la diffusion excessive des IntPK sur le réseau véhiculaires, on a ajouté un compteur de sauts dans l'en-tête LAL qui est incrémenté à chaque saut. Un IntPK est abandonné lorsque son nombre de sauts dépasse une limite prédéfinie.

### 4.3.3 Discussion

#### 4.3.3.1 Avantages

- La stratégie implémentée au niveau du CS, consistant à la conservation de toutes les données reçues que ce soit à travers des DataPK ou simplement à travers les mouvements des véhicules, peut être avantageuse en termes de temps de réponse, et cela en permettant une diffusion rapide surtout dans un environnement fortement dynamique ;
- Lorsqu'une voiture répond à un intérêt avec des données, la réponse peut se propager par le canal de diffusion sans fil aux voitures voisines et être mise en cache par tous les récepteurs. Lorsque ces voitures se déplacent physiquement, ils vont servir de Data-mule transportant les données à une plus grande zone ;
- Le fait d'avoir un grand nombre de Data-mule augmente les zones de diffusion des données et donc la probabilité de croisement des véhicules consommateurs avec les véhicules qui portent des copies des données recherchées ;
- Cette architecture permet le multi-homing, car toutes les interfaces peuvent être utilisées en même temps lorsqu'il est nécessaire ;
- La sélection du véhicule le plus éloigné pour transmettre les IntPK, peut limiter le problème de Broadcast storm flooding.

#### 4.3.3.2 Inconvénients

- Il est difficile voire impossible de créer ou maintenir des entrées fixes dans la FIB, en raison de la nature dynamique de l'environnement VANET, cette contrainte nous force obligé à diffuser les IntPK vers toutes les interfaces qui sont disponibles au lieu d'utiliser des protocoles de routage ;
- L'absence des mécanismes intelligents (i.e. protocoles de routage) qui renforce et assure une transmission de paquets sans gaspillage des ressources de transmission ;
- Contrairement aux réseaux de capteurs (WSN) et aux téléphones mobiles, les véhicules se disposent d'une grande capacité de stockage de données, mais il est quand même nécessaire de choisir quelle donnée faut-il conserver dans le cache pour ne pas gaspiller les ressources de stockage (parmi les éventuelles solutions : on peut

ajouter une durée de vie aux données sauvegardées, et à l'expiration de cette durée la donnée sera supprimée du CS);

- La stratégie de transfert de données utilisées augmente le temps de réponse, ce qui dégrade la performance du système.

## 4.4 Le 2<sup>ieme</sup> design : Navigo, Transfert de données en utilisant les positions géographiques

Le défi fondamental dans la conception de Navigo [18] est de savoir comment orienter les IntPK vers la localisation des producteurs de données. En effet, la connectivité intermittente dans l'environnement véhiculaire dynamique rend l'exécution des protocoles de routage difficile. Cependant, les auteurs de [ref-doc2-Navigo -21] ont démontré que pour les applications de trafic qui sont intrinsèquement liées aux informations de localisation (i.e. c'est-à-dire que les données produites et consommées par celles-ci sont liées à des emplacements spécifiques, ces emplacements sont intégrés dans les noms de données), il est possible de transférer simplement les IntPK vers les positions géographiques indiquées dans les noms, sans avoir besoin d'un protocole de routage. Mais, pour comprendre la sémantique des noms et pouvoir extraire les informations de destination, il faut néanmoins implémenter dans chaque nœud une stratégie intelligente de transfert de données. Une telle hypothèse peut être bénéfique au réseau VNDN mais elle n'est pas encore réalisable dans le cadre actuel de NDN, où la stratégie de transmission ignore la sémantique des noms. Donc, en présence de cette contrainte les consommateurs ne sont pas capables de préciser ou de suggérer les endroits où les données peuvent résider. En outre, d'autres types d'applications, comme le partage de musique ou l'acquisition de données en général, ne sont pas associés à des emplacements spécifiques.

Pour transmettre efficacement les IntPK vers tous les types d'applications sans protocole de routage, les auteurs de [ref-doc2-Navigo] ont proposé une solution qui consiste à associer les noms de données aux emplacements géographiques où résident les données :

- Pour le premier type d'applications (qui dépendent des informations de localisation), ils ont lié les noms aux emplacements sélectionnés par le consommateur (un segment de route, parking, etc);
- pour le deuxième type d'applications (qui ne dépendent pas des informations de localisation), leur solution était de lier les noms de données aux emplacements des véhicules qui possèdent ces données, ils peuvent être des producteurs originaux ou des Data-mules qui portent les données dans leurs caches, ou des points d'accès Internet (RSU).

### 4.4.1 Dénomination des zones géographiques

Dans cette architecture, la carte est divisée en régions selon le Système de référence de grille militaire (the Military Grid Reference System - MGRS). Ce système est dérivé du système Universal Transverse Mercator (UTM) et des réseaux de grille Universal Polar Stereographic (UPS), où chaque région est identifiée par une étiquette (label).



FIGURE 4.5 – La carte MGRS [18].

Dans l'implémentation actuelle, les zones géographiques ont une taille fixe (200x200 mètres), mais le schéma MGRS [voir figure 4.5] permet facilement d'ajouter des précisions différentes, en ajoutant ou en supprimant des chiffres du nom, par exemple : 4QFJ 12 67 définit une précision de 1Km, alors que QFJ 123 678 définit une précision de 100 mètres.

### 4.4.2 Cartographie des noms de données

Lorsqu'un nœud veut envoyer un IntPK pour rechercher des données qui n'ont aucune signification géographique et qu'il n'a aucune connaissance du préfixe du nom de données (pas dans sa FIB), le nœud envoie simplement l'IntPK dans toutes les directions :

- Si l'un de ces IntPK inondés atteint une copie des données correspondantes, le répondeur attache son nom de région géospatiale MGRS (par exemple, 4QFJ 123 678) au DataPK renvoyé ;
- en retour vers le nœud demandeur le DataPK suit la trace du IntPK, tous les nœuds qui le reçoivent apprennent la liaison entre le préfixe de nom de données et la zone géographique correspondante ;

- cette information leur permettent de transmettre directement les futurs IntPK qui portent le même préfixe de nom à cette zone géographique uniquement ;
- pour les données dépendantes de la localisation, le consommateur peut éviter la procédure initiale d'inondation d'intérêt en liant le nom de Données à la zone géographique à laquelle le consommateur est intéressé avant d'envoyer l'Intérêt.

### 4.4.3 Architecture Navigo

Le démon NDN actuel ne comprend pas le concept des zones géographiques. En effet, la FIB ne contient que des paires (préfixe, face), donc pour faire face à cette contrainte et pour pouvoir exploiter la liaison entre les noms de données et les zones géographiques, Navigo a introduit un nouveau concept à l'architecture VNDN actuelle. Ce concept est appelé le Geographic Faces (GeoFaces) [voir figure 4.6].

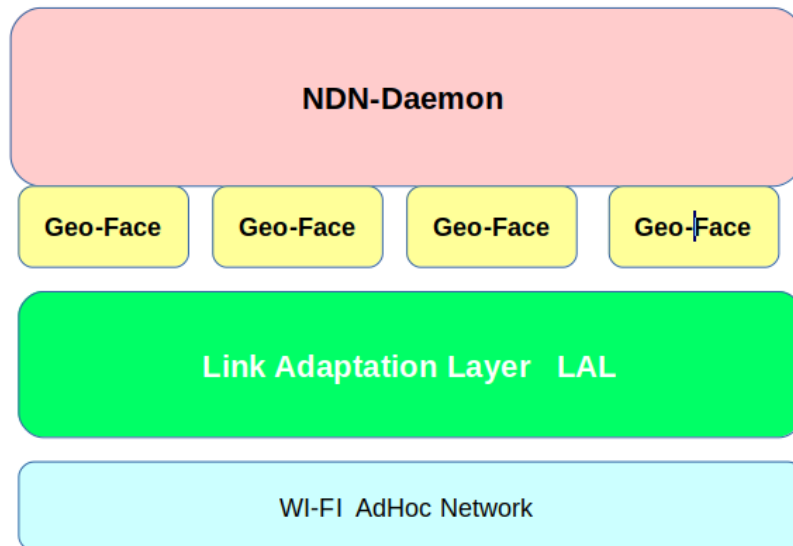


FIGURE 4.6 – Architecture VNDN-Navigo[18].

La liaison des préfixes de nom aux zones géographiques se fait donc en deux étapes : on lie d'abord la zone géographique à un GeoFace, puis on laisse au FIB de stocker la liaison (le mappage) entre les noms de données et GeoFaces comme dans l'architecture NDN actuelle de manière à permettre au consommateur d'enregistrer simplement une nouvelle règle dans la FIB pour lier un préfixe à une zone géographique spécifique.

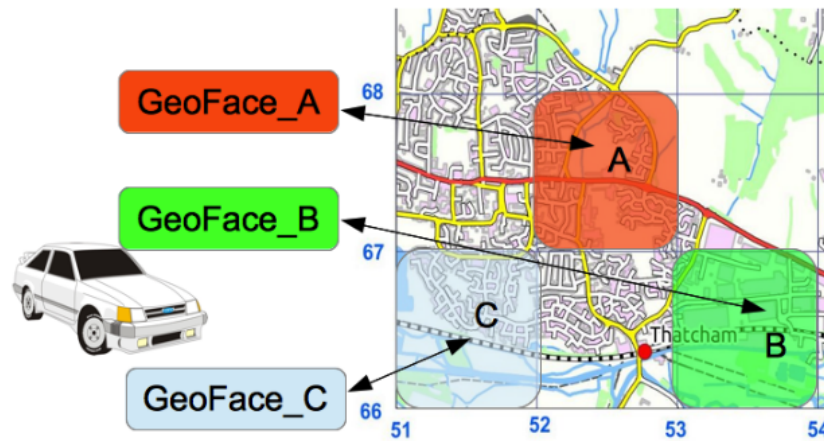


FIGURE 4.7 – Mappage GeoFaces – zone géographique[18].

Le GeoFace A [voir figure 4.7] ne peut envoyer des paquets que vers la zone géographique A, et ne peut également recevoir des données qu'à partir de cette dernière.

#### 4.4.4 l'intégration des informations géographiques dans la pile NDN

Ce design est axé sur les scénarios urbains. On suppose que chaque véhicule est équipé d'un capteur GPS et d'une carte numérique, lui permettant d'identifier sa propre position géographique (et donc son nom MGRS). On suppose qu'il est également équipé d'une interface de communication Wi-Fi Ad-Hoc et une grande capacités de stockage et de calcul. En outre, on suppose la présence de RSU dans tous les segments routiers pour assurer la connexion Internet aux véhicules intégrant la pile NDN et Navigo.

L'idée principale derrière la stratégie de transfert de données adoptée par Navigo est d'explorer la zone entourant le nœud lors de la recherche des producteurs, des Data-mules ou des RSU, ensuite, dès qu'on reçoit le premier DataPK, on enregistre la liaison entre le prefix de nom et sa géolocalisation pour pouvoir transmettre les futurs IntPK vers la même zone géographique. On résume cette stratégie comme suit : soit  $i$  un IntPK, et  $N$  la donnée recherchée,

- Si  $i$  ne correspond à aucune entrée dans la FIB (il n'y a aucune information sur l'emplacement des données), il sera envoyé en utilisant une technique d'inondation sans spécifier la zone de destination (le nœud est donc dans la phase d'exploration) ;
- si plusieurs GeoFaces sont liées à  $N$  dans la FIB, le nœud sélectionne les interfaces suivant la technique de round-robin ;
- si une seule interface est disponible, avec la probabilité  $P$ , le GeoFace stocké dans la FIB sera utilisé, sinon si elle est disponible avec la probabilité  $(1 - P)$  le nœud agit comme dans la phase d'exploration, (il inondant l'IntPK) ;

- Navigo adopte cette phase d'exploration supplémentaire pour éviter de se concentrer sur une seule zone de destination pendant une longue période de temps ;
- Après avoir envoyé le IntPK, si le nœud ne reçoit pas les données demandées avant l'expiration d'un délai T (timer, par exemple : 300 ms), la liaison dans la FIB entre le GeoFace et N sera supprimée ;
- Si plusieurs IntPK  $i$  portant le même préfixe N, sont envoyés en pipeline en utilisant le même GeoFace F : alors dès que l'un d'eux est satisfait, le timer T sera supprimée pour tous les IntPK en attente pour N, car la réception d'un DataPK signifie que l'information est encore disponible dans cette zone géographique, même si certains des IntPK envoyés échouent. [voir figure 4.8]

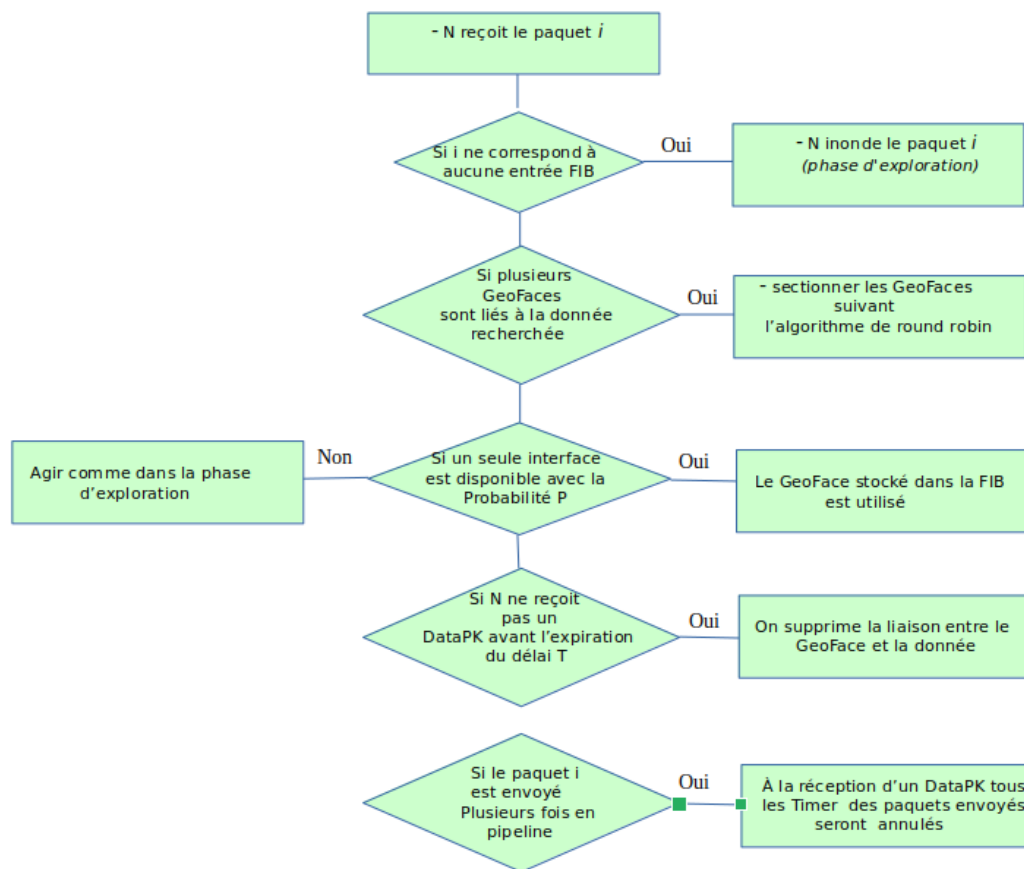


FIGURE 4.8 – La stratégie de transfert de données adoptée par Navigo[18].

#### 4.4.5 Gestion de la FIB

Les noms dans NDN sont hiérarchiques (par exemple le nom  $/N$  peut être divisé en plusieurs morceaux, c'est-à-dire  $/N/m1$  ;  $/N/m2$ , etc). Pour transférer correctement les IntPK de tous ces morceaux (et seulement eux), il faut avoir une entrée FIB avec le préfixe  $/N$ . En effet, Le consommateur et le producteur connaissent la sémantique des

noms et peuvent donc identifier quel préfixe agrège toutes les pièces d'un contenu. Mais les nœuds intermédiaires (Forwarders), ne peuvent pas exécuter l'application, car ils ne sont pas au courant de ces noms et ne peuvent pas donc remplir correctement la FIB. La couche LAL du consommateur peut résoudre ce problème en attachant à chaque IntPK le préfixe qui regroupe tous les éléments du contenu demandé. Cette information est ensuite diffusée par les nœuds intermédiaires, ce qui permet à chaque LAL d'enregistrer la règle correcte dans la FIB chaque fois que l'IntPK est satisfait.

#### 4.4.5.1 La taille de la FIB

la FIB de l'architecture Navigo crée des entrées qui stocke les préfixes de noms en les reliant aux GeoFaces qui peuvent être utilisés pour récupérer les données correspondantes aux préfixes. En effet, l'association des zones géographiques avec des GeoFaces, peut affecter la taille de la FIB, plus la dimension de la zone géographique est importante plus la taille de la FIB est grande et vice-versa.

La probabilité de réception des données peut par ailleurs être affectée par la dimension de la zone, plus la dimension est réduite plus la probabilité de recevoir un DataPK de différentes régions augmente. Et inversement la probabilité diminue si la zone géographique est de grande dimension.

#### 4.4.6 La couche Link Adaption Layer (LAL)

Navigo étend la version originale de la couche LAL présentée dans le premier design [section 1] tout en exploitant les informations géographiques de la zone de destination. elle dirige le IntPK vers le plus court chemin en tenant compte des scénarios urbains et de la présence des obstacles. Elle se charge aussi de la création et la destruction des GeoFaces en conservant le mappage entre GeoFaces et les zones géographiques dans une structure de données appelée Face to Area table (F2A) :

- À la réception d'un DataPK, le LAL extrait les informations géographiques de la zone fournies par le producteur de données ;
- Si cette zone n'est associée à aucun GeoFace, la LAL crée un nouveau GeoFace et lie ce dernier à la zone (en ajoutant la relation à la F2A) ;
- La couche LAL vérifie périodiquement la table F2A (chaque dizaines de secondes) et purge les entrées inutilisées (elle purge également la FIB).

##### 4.4.6.1 Calcul du chemin le plus court

Le chemin le plus court vers la zone de destination est calculé en appliquant l'algorithme de Dijkstra, dont les segments de routes représentent les arêtes du graphe tandis

que les intersections représentent les nœuds. (Les routes principales sont les plus prioritaires). Dans la [figure 4.9] le véhicule V2 va transmettre l'IntPK car il est le plus proche vers le producteur V4

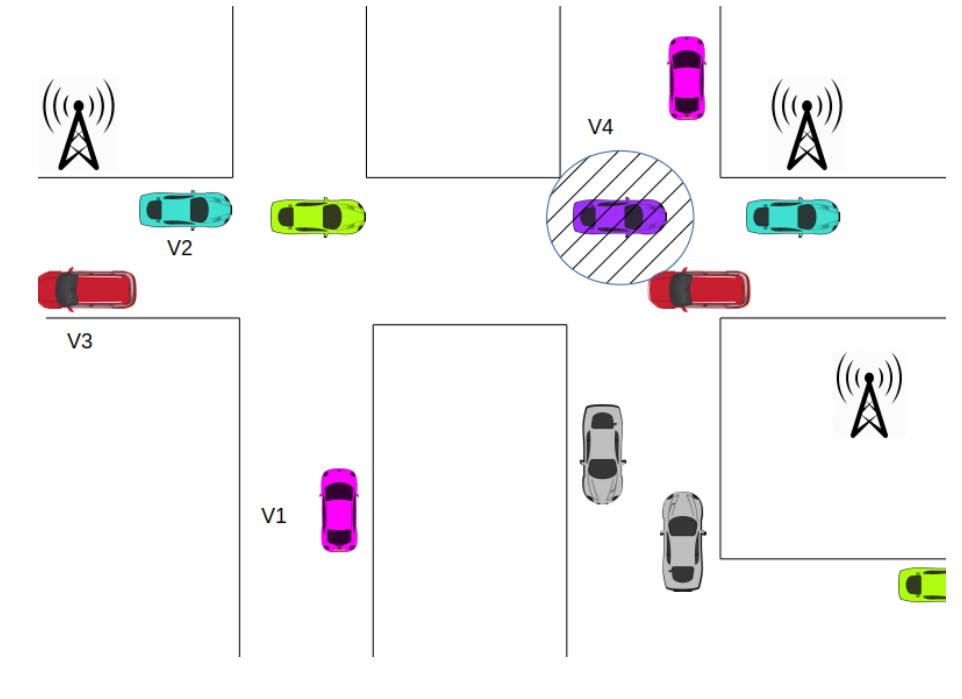


FIGURE 4.9 – Calcul du plus court chemin.[18]

#### 4.4.7 Le processus de transfert de données

- Lorsque le NDN daemon envoie un IntPK via un GeoFace, ce dernier passe le paquet à la couche LAL, [voir figure 4.10];
- la LAL effectue une recherche dans la table F2A pour déterminer le nom de la zone de destination;
- cette information est ensuite codée dans un en-tête qui encapsule le IntPK avec la position du nœud;
- ensuite elle diffuse l'information vers tous les voisins dans la plage de transmission;
- dès qu'un véhicule reçoit le paquet, la LAL de ce véhicule extrait et stocke les informations géographiques de la zone de destination, la position du nœud précédent et la nonce du IntPK, puis elle passe le paquet au NDN daemon;
- Si le NDN daemon décide de transférer le paquet vers le réseau V2V, il le passe vers la LAL, soit en utilisant GeoFace spécifié dans la FIB, soit en utilisant la procédure de phase d'exploration (vue dans la section 4.4.4);
- en se basant sur la nonce de l'IntPK, la LAL, récupère la position du nœud précédent et la zone de destination spécifiée par le consommateur originaire du paquet;

- la LAL calcule e plus court chemin vers la destination et transmet le IntPK (si elle est plus est plus proche à la zone de destination par rapport au nœud précédent) ;
- une fois que le IntPK atteint la zone de destination, Navigo utilise le protocole décrit dans [ref-doc1-VNDN- 20] pour effectuer une diffusion local, en inondant le paquet dans toutes les directions disponibles.
- les voitures situées à l'extérieur de la zone de destination peuvent répondre à la requête, mais elles ne diffusent plus le paquet, (Navigo limite la diffusion des IntPK dans la zone de destination uniquement) ;
- lorsqu'un IntPK atteint le producteur de données, le nœud va répondre par un DataPK, en joignant son nom MGRS à l'en-tête de la couches 2,5 ;
- tel que défini par le protocole NDN, les DataPK suivent en retour le chemin inverse du IntPK en sautant d'une PIT à une autre jusqu'à revenir au consommateur ;
- la LAL à chaque fois met à jour la FIB en liant le préfixe de nom au GeoFace associé à la zone géographique de la donnée et transmet la paquet si elle est plus proche à la destination par rapport au nœud précédent à partir duquel elle a reçu le IntPK ;
- pour augmenter la fiabilité des communications, la LAL utilise le concept d'accusé de réception implicite introduit par [18].

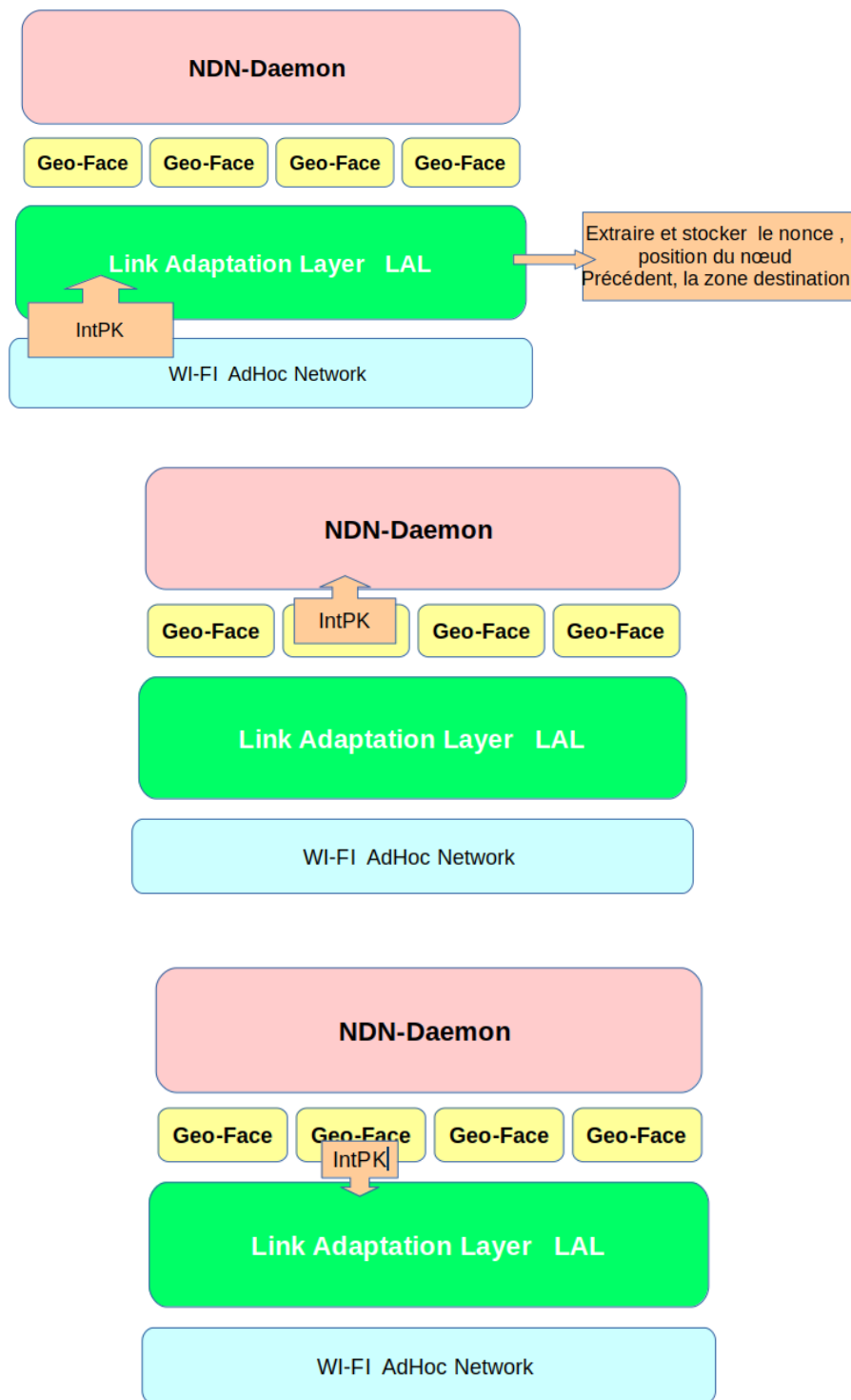


FIGURE 4.10 – Le processus de transfert de données[19].

En raison de la présence des obstacles dans les zones urbaines, il est difficile de couvrir une grande zone géographique avec un plus petit nombre de sauts. c'est pour cela Navigo a implémenté une stratégie de retardement de transmission des paquets en fonction des

positions géographiques des véhicules et les types des paquets à transmettre :

- Les DataPK sont plus prioritaires que les IntPK ;
- les voitures situées aux intersections sont plus prioritaires que les autres ;
- le véhicule le plus loin du dernier saut (nœud précédent), va attendre le moins de temps possible, et vice versa. Cette technique est utilisée pour éviter les collisions.

## 4.4.8 Discussion

### 4.4.8.1 Avantages

- L'intégration des positions géographiques dans les données permet d'effectuer des redirections géographiques pour prendre en charge tous les types d'applications ;
- les applications peuvent lier les noms de données aux GeoFaces pour faciliter la recherche et le transfert de ces dernières ;
- le design Navigo a profité de la nature dynamique de l'environnement véhiculaire, plus particulièrement dans les zones géographiques de petites dimensions, car le déplacement rapide des producteurs de données va augmenter le nombre des GeoFaces qui mènent aux mêmes préfixes de nom, et par conséquent la probabilité de recevoir un DataPK augmente aussi.

### 4.4.8.2 Inconvénients

- La stratégie d'inondation dans la phase d'exploration peut causer un problème de collision dans les zones.

## 4.5 Le 3<sup>ieme</sup> design : Architecture VNDN hiérarchique et dynamique

D'autant que l'adoption des clusters est une technique assez courante en ce qui concerne l'organisation des véhicules dans les réseaux VANET, les auteurs de [ref-doc3-HVNDN] ont introduit cette technique aux VNDN en tant que support de plusieurs fonctionnalités fournies par NDN.

Deux aspects clés caractérisent le modèle proposé par le groupe [20] :

- tout d'abord, on définit un VNDN hiérarchique structuré de manière à permettre au réseau et aux couches véhiculaires supporter la recherche, l'acheminement et le routage, tout en réduisant la diffusion d'informations et l'accès aux infrastructures routières ;

- deuxièmement, les véhicules sont dynamiquement regroupés et utilisent des communications hybrides, à courte portée pour V2V et cellulaires pour V2I, où le réseau cellulaire agit comme de la colle pour la gestion du réseau VNDN.

### 4.5.1 Architecture du réseau

Cette architecture est organisée en quatre différentes couches [voir figure 4.11] , dont les deux couches inférieures interagissent directement avec l'environnement véhiculaire, tandis que les deux couches supérieures prennent en charge les différentes fonctions des infrastructures du réseau. La figure ci dessous montre les différentes couches de cette architecture.

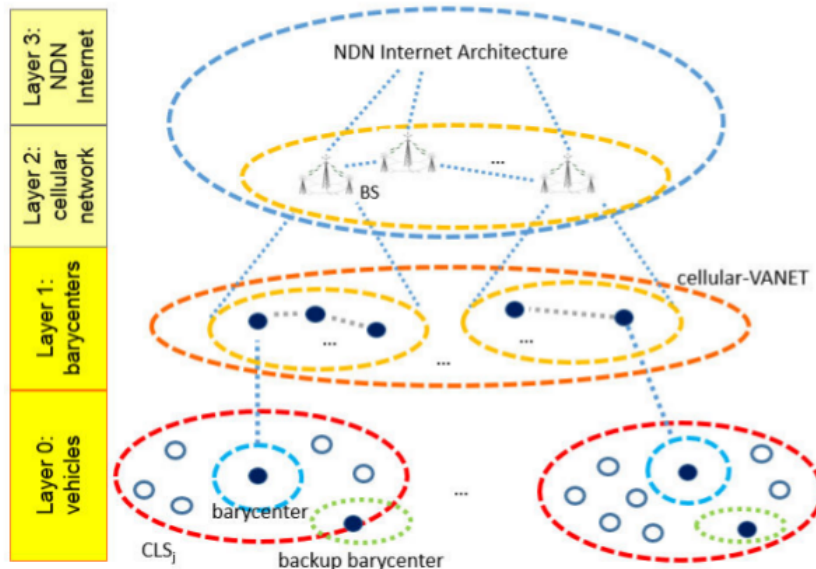


FIGURE 4.11 – Architecture hiérarchique VNDN à quatre couches[20].

#### 4.5.1.1 La couche 0

Les véhicules sont regroupés en clusters, dans chacun on trouve un sous ensemble de véhicules :

- Véhicules de référence (Barycenters) : le rôle de ces véhicules de référence est similaire à celui des têtes de cluster dans les réseau de capteurs (WSN), ils sont chargés de communiquer avec les couches cellulaires, ainsi que de rechercher les informations dans le cluster et de les acheminer aux demandeurs, c'est à dire qu'ils servent de référence pour les requêtes originaire de leurs clusters et fournissent au même temps des fonctionnalités de consultation aux autres clusters .( on ne trouve qu'un seul Barycenter par cluster ) ;

- Barycenters de secours (Back-Up Barycenters) : c'est le deuxième sous ensemble qu'on peut trouver au niveau de cette couche. Un Back-Up Barycenter prend la place d'un Barycenter si ce dernier n'est plus visible par les véhicules de son cluster.

#### 4.5.1.2 La couche 1

Regroupe tous les Barycenters du réseau; chaque Barycenter utilise deux types de communication :

- Communication V2V : entre les Barycenters de la cellule, et entre les Barycenters les véhicules de leurs clusters ;
- communication V2I : pour communiquer avec la couche cellulaire.

Les Barycenters sont eux même regroupés en cellules, où chacune correspond à une station de base (BS) de la couche supérieure.

#### 4.5.1.3 La couche 2

Au niveau de cette couche on trouve un réseau cellulaire qui regroupe toutes les BS du réseau qui possèdent les caractéristiques suivantes :

- Chaque BS fait référence à une zone bien déterminée ;
- chaque BS sert un nombre variable de clusters ;
- une BS peut rendre la communication entre des Barycenters de différentes cellules possible.

#### 4.5.1.4 La couche 3

le réseau internet NDN est la dernière couche dans cet architecture, elle est accessible par le réseau cellulaire.

### 4.5.2 Élection des Barycenters

Avant de décrire la procédure d'élection d'un Barycenter dans un cluster qu'appartient à une cellule référencée par une BS, on doit tout d'abord introduire la liste de messages suivante :

#### 4.5.2.1 La procédure d'initialisation

Une cellule est un groupement de Barycenters gérés et coordonnés par une BS qui fait référence à une zone géographique, noté (S); cette zone S contient un nombre variable de clusters, noté (Nc). Lorsqu'un cluster décide qu'il est nécessaire d'élire un nouveau barycentre :

- La BS diffuse le message INV dans la zone de S (cellule);

INV (Invite)	message envoyé par le réseau cellulaire (BS) pour inviter des véhicules d'un cluster à jouer le rôle d'un barycentres
RESB (Response Barycenter)	réponse d'un véhicule disponible dans un cluster référencé par la BS qui a envoyé le message INV, et qui a accepté le rôle de Barycentre
DEFB (define Barycenter)	définition du Barycentre
DEFBB (define Back-Up Barycenter)	définition du Back-Up Barycentre
BARON	Barycenter ON
ATT (Attention)	demande d'attention

- et les véhicules qui reçoivent ce message vont déclarer leur disponibilité pour servir de barycentres à travers le message RESB ;
- si la BS reçoit plusieurs messages RESB, alors elle peut choisir le barycentre selon des règles distinctes, comme par exemple le choix du véhicule le plus éloigné, ou même elle peut procéder d'une manière aléatoire ;
- la BS désigne ensuite le Barycentre éligible en utilisant DEFB et le Back-Up Barycentre en utilisant le message DEFBB ;
- en fin, la cellule (BS) cesse de diffuser des messages d'invitation INV, et le barycentre élu diffuse périodiquement sa disponibilité en utilisant le message BARON qui contient ses coordonnées GPS pour fournir des services NDN.

#### 4.5.2.2 La Réélection d'un Barycenter

L'architecture proposée dans [20] évolue avec le temps en raison de la mobilité des véhicules, par conséquent un barycentre ne peut pas continuer à jouer son rôle s'il n'est plus visible par les véhicules de son cluster (i.e. éteint, hors de la zone du cluster/cellule, etc). Dans ce cas le barycentre doit être réélu suivant le même processus décrit précédemment. Et entre-temps le Back-Up Barycenter prend la place de l'ancien Barycenter qui est hors service et diffuse son nouveau rôle dans le cluster accompagnés avec ses coordonnées GPS.

#### 4.5.3 Architecture des Barycenters

Pour pouvoir supporter les opérations NDN, chaque barycentre doit maintenir a une structure contenant trois modules [voir figure 4.12] :

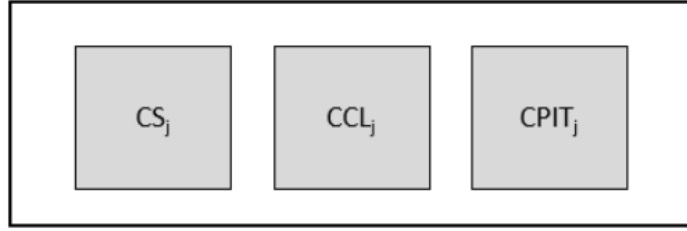


FIGURE 4.12 – Les modules du Barycenter[20].

- **Content Store** ( $CS_j$ ) : c'est une mémoire cache locale au niveau du Barycenter ;
- **Cluster Contents List** ( $CCL_j$ ) : c'est une liste qui associe les données aux véhicules qui la possèdent dans le cluster ainsi que leurs positions géographiques ;

$$CCL_j = (name_{kj}, GPS(v_i))$$

Où :

$$j \in 1 \dots CLS_{tot}, k \in 1 \dots CNT_{tot}, i \in 1 \dots NV_{tot}, v_i \in CLS_j$$

$CLS_{tot}$  = nombre total des clusters dans la cellule,  $CNT_{tot}$  = le nombre de contenus distincts disponibles dans toute la structure ,  $NV_{tot}$  = Le nombre total de véhicules à l'instant t,  $CLS_j$  = le cluster j ,  $j \in 1 \dots CLS_{tot}$  ;  $v_i$  = un véhicule de la couche 0.

- **Cluster Pending Interest Table** ( $CPIT_j$ ) : garde la trace de tous les IntPK des véhicules du cluster.

$$CPIT_j = (name_{kj}, GPS(v_{req})),$$

Où :

$$j \in 1 \dots CLS_{tot}, k \in 1 \dots CNT_{tot}, req \in 1 \dots NV_{tot}, v_i \in CLS_j, Req = requête$$

#### 4.5.4 Modèle de communication hiérarchique

Lorsqu'un véhicule  $v_i$  de la couche 0, est à la recherche d'une donnée identifié par son nom  $name_{kj}$ , il envoie directement son IntPK au Barycenter responsable du cluster pour éviter l'inondation de ce paquet dans tout le réseau. Le processus de récupération de la donnée recherchée peut s'exécute au pire en quatre étapes :

- Recherche intra-cluster, au sein d'un cluster (niveau 0) ;
- recherche inter-cluster, parmi les grappes d'une cellule (niveau 1) ;
- recherche dans les clusters des cellules distinctes (niveau 2) ;

- Recherche dans Internet-NDN (niveau 3).

#### 4.5.4.1 Recherche intra-cluster

Lorsqu'un IntPK portant le nom  $name_{kj}$  atteint Barycenter :

- Le nom est d'abord recherché dans  $CS_j$ , dans le cas où il est trouvé, la données sera transmise au consommateur ;
- Si c'est pas le cas, autrement dit, si le  $CS_j$  ne dispose pas de la donnée recherchée, le  $CCL_j$  est recherché, dans le cas où le  $name_{kj}$  est trouvé, la donnée sera récupérée du premier véhicule répertorié (producteur), ensuite retourné au Barycenter ;
- à la réception d'un DataPK par le Barycenter, il va tout d'abord stocker la donnée dans son  $CS_j$  ensuite la transférer vers le consommateur ;
- Dans le cas contraire si le  $name_{kj}$  n'appartient pas au  $CCL_j$ , il sera recherché finalement dans la  $CPIT_j$ , dans le cas où le nom est présent dans la liste, la requête sera mise en file d'attente dans  $PIT_j$  ;
- requête sera envoyée en dehors du cluster autrement.

Le processus décrit au dessus est montré dans la [voir figure 4.13] :

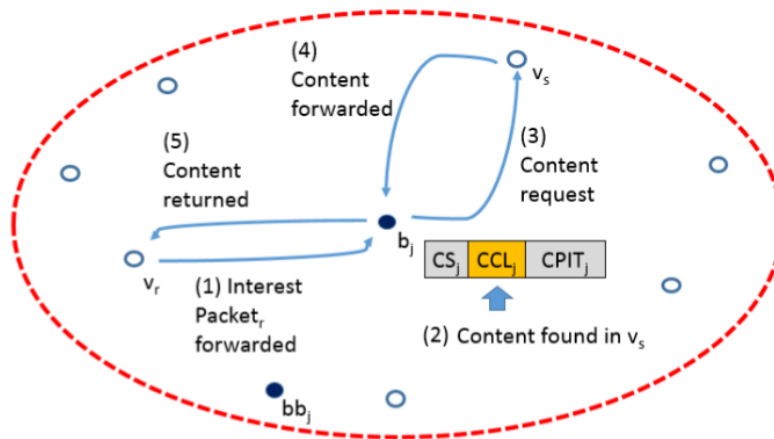


FIGURE 4.13 – Recherche intra-cluster[20].

(1)  $v_r$  diffuse un IntPK à Barycenter  $b_j$ , (2) la donnée n'est pas trouvée dans  $CS_j$ , mais  $CCL_j$  indique qu'il est en  $v_s$ , dans le cluster. (3) La demande est donc envoyée à  $v_s$ , (4) qui transmet la données souhaité à  $b_j$ , (5) et finalement  $b_j$ , la renvoie à  $v_r$ .

#### 4.5.4.2 Recherche inter-cluster

Dans le cas où le IntPK n'est pas dans  $PIT_j$ , Barycenter transmet la requête à d'autres Barycentres de la cellule. Puisque les PIT empêchent les retransmissions, une seule transmission par barycentre est effectuée :

- Si le IntPK est dans un  $CS_q$ , la donnée sera transmise récursivement vers le Barycenter originaire de la requête ;
- si c'est pas le cas, les  $CCL_q$  sont recherchées ; dans le cas où le nom  $name_{kj}$  est trouvé dans une  $CCL_q$ , c'est-à-dire dans l'un des véhicules de la couche 0, un DataPK est renvoyé récursivement au consommateur en passant par tous les barycentres impliqués ;
- dans le cas où le nom d'un IntPK n'est pas dans une  $CCL_q$ , mais il est dans une  $PIT_q$ , cette dernière sera mise à jour avec la nouvelle requête ;
- dans chaque type de recherche inter-cluster, à la fois à l'intérieur d'une cellule ou parmi d'autres cellules, le nombre maximal de transmissions entre véhicules est linéaire avec le nombre de barycentres impliqués, puisque les PIT empêchent la même requête d'être transmise deux fois ou plus.

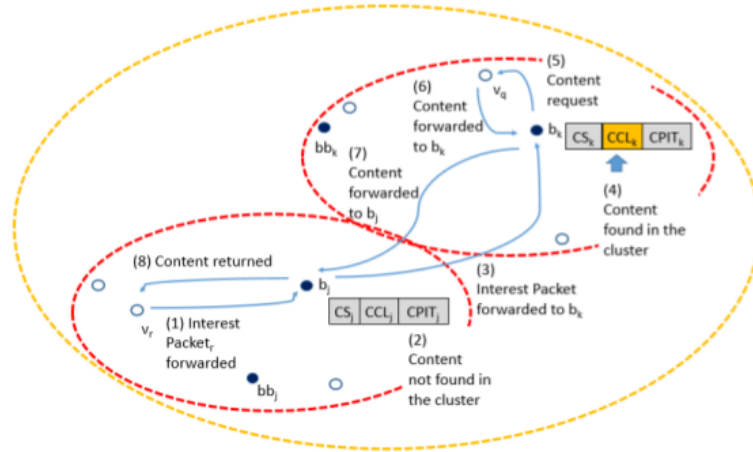


FIGURE 4.14 – Recherche inter-cluster[20].

Dans la [figure 4.14] : (1)  $v_r$  envoie un IntPK à  $b_j$ , (2) qui ne la trouve dans aucun de ses modules ; (3) La requête est ainsi transmise à d'autres barycentres, (4) parmi lesquels  $b_k$  qui trouve le contenu dans  $CCL_k$ , en particulier dans  $v_q$ , (5)  $b_k$  envoie un IntPK à  $v_q$ . (6) qui répond par un DataPK, (7) ensuite le DataPK sera transmis à  $b_j$ , (8) et finalement retourné à  $v_r$ .

#### 4.5.4.3 Recherche dans les clusters des cellules distinctes

Si la recherche inter-cluster échoue, autrement dit si la données demandée n'est pas disponible dans la cellule, alors tout le réseau cellulaire sera impliqué dans le processus de recherche, afin de déterminer quelle cellule ou cellules doivent être accédées.

#### 4.5.4.4 Recherche dans Internet-NDN

Si la recherche effectuée au troisième niveau de hiérarchie échoue encore, alors la donnée demandée n'est pas disponible dans tout le réseau VANET, et on doit donc la rechercher dans Internet-NDN.

### 4.5.5 Discussion

#### 4.5.5.1 Avantages

- La structure hiérarchique permet de réduire le nombre d'accès à Internet, en recherchant les données aux différents niveaux de l'architecture ;
- dans le cas favorable, c'est à dire dans le cas où on trouve la donnée souhaitée au niveau 0 de hiérarchie, on aura une meilleure utilisation des ressources de transmission ;
- permet de réduire le problème de collision à l'intérieur des clusters, car les requêtes seront directement transmises vers les Barycenters, et c'est à lui de les transférer vers les véhicules producteurs ;
- le réseau cellulaire empêche les Barycenters de devenir des points de congestion ;
- l'organisation dynamique et scalable, rend le désigne flexible et augmente les potentialités du système.

#### 4.5.5.2 Inconvénients

- Pour que ce désigne soit adapté au réseau VNDN, il est indispensable que tous les véhicules soient équipés d'une technologie compatible avec le réseau cellulaire ;
- les véhicules situés aux extrémités des clusters peuvent causer un problème de collision entre les clusters adjacents de la même cellule ;
- il faut utiliser des périodes Silence-parole gérées par les BS pour éviter la collision entre les clusters adjacents de la même cellule, et pour se faire il faudra que tous les véhicules soient équipés d'un GPS pour pouvoir déterminer leurs positions dans la zone S, et les classer ensuite dans l'ensemble des silence/parole ;
- on peut également résoudre le problème de collision entre les clusters, en utilisant des fréquences différentes ;
- pour que les différents Barycenters d'une même cellule puissent communiquer entre eux, il faut que la taille des clusters soit toujours inférieure ou égale au rayon (R) prédéfinie ;
- ce désigne peut par ailleurs augmenter le temps de bout en bout ce qui dégrade la performance du système.

## 4.6 Le 4<sub>ieme</sub> design : Vehicular Named Data Networking VEDNET

Dans le réseau véhicule-infrastructure (V2I), les assistances de transmission sont vivement nécessaires (i.e. 2.5G, 3G, 4G), pour gérer de manière centralisée toute les communications entre véhicules. Avec la technique de handover entre les cellules radio, les véhicules suivent toujours le rythme du serveur qui fournit les applications VANET. Par conséquent, le handover entre les stations de base est inévitable car les véhicules sont souvent en mouvement. C'est pour cette raison que le routage proactif est utilisé dans le réseau V2I. Les auteurs de VEDNET [21] ont proposé l'utilisation du dernier réseau cellulaire : Long Term Evolution (LTE) pour les réseaux V2I.

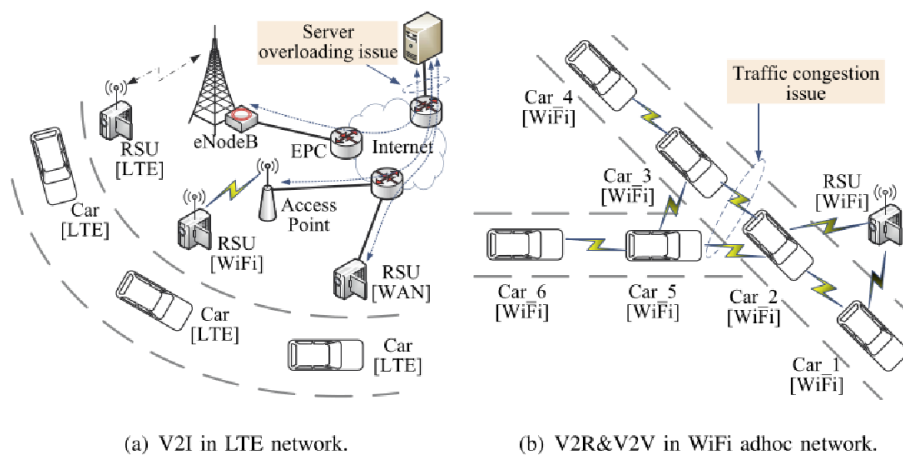


FIGURE 4.15 – Deux aspects de la communication des véhicules [21].

La [figure 4.15.(a)] Montre un exemple de communication V2I dans le réseau LTE. Les unités routières (RSU) sont des caméras pour capturer l'état de la circulation routière. En outre, les connexions aux RSU peuvent être faites via LTE, Wireless Fidelity (WiFi) ou Wide Area Network (WAN). Tandis que les véhicules demandent et reçoivent les mises à jour de l'état du trafic routier depuis le serveur via LTE.

[La Figure 4.15.(b)] montre un exemple de communication V2R et V2V dans un réseau ad-hoc WiFi : le véhicule (Car-2) et RSU rencontrent un problème de congestion de trafic lorsqu'un grand nombre de véhicules sont proches du RSU et demandent tous en même temps le même contenu. Dans ce cas, le canal WiFi devient un goulot d'étranglement car tous les nœuds sans fil partagent la même bande passante de communication.

### 4.6.1 le routage réactif dans NDN

Dans le routage réactif, les nœuds ne disposent pas de table de routage (FIB dans le mécanisme NDN). Donc pour configurer un chemin inverse du serveur vers un utilisateur final, l'inondation est la seule solution car elle permet d'implémenter la diffusion multi-hop des IntPK. Cependant, le mécanisme de diffusion entraîne plusieurs problèmes :

- Une transmission en rafale peut se produire en diffusant tous les IntPK reçus. L'inondation dans de nombreux cas, notamment dans les réseaux denses, peut causer un overhead de communication à cause de la rediffusion redondante ;
- un problème de boucle de routage se produit lorsque plus de deux nœuds intermédiaires disposent de la même fréquence radio ;
- un problème de Broadcast Storm est généré lorsqu'il existe plusieurs chemins inverses du serveur vers le client. Les auteurs de VEDNET ont suggéré de minimiser le nombre de sauts entre le RSU et le véhicule cible, pour pouvoir restreindre le nombre de nœuds reliant les données de diffusion sans avoir besoin d'utiliser d'autres information..

Le calcul du nombre minimal de sauts se fait par les nœuds intermédiaires avant la diffusion, en vérifiant la nombre de sauts intégrés dans les IntPK reçus. Généralement, le premier IntPK reçu a toujours un nombre de sauts minimum et sera diffusé. Ensuite, tous les IntPK arrivés plus tard et qui demandent la même donnée devraient être supprimés. La [Figure] montre les principales opérations faites par les véhicules NDN.



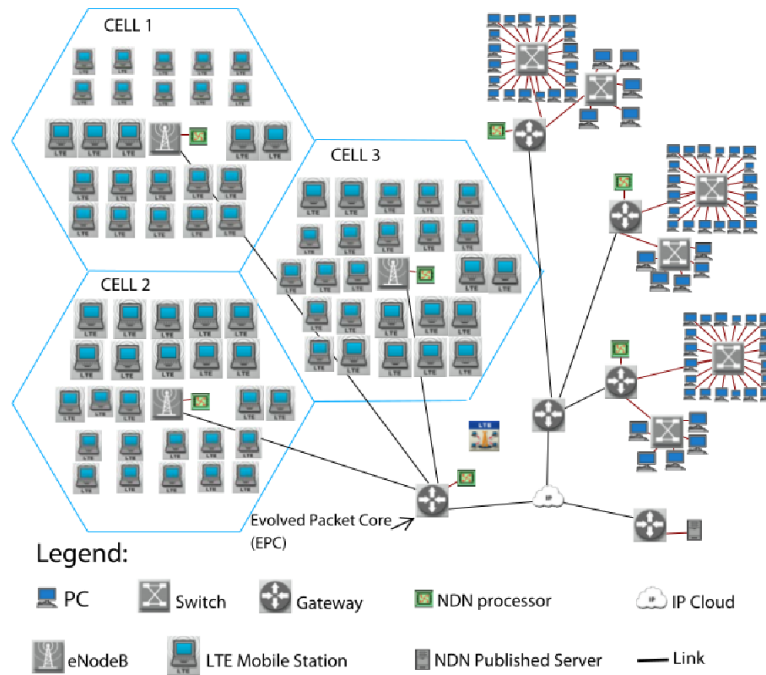


FIGURE 4.17 – Architecture VEDNET[21].

La même configuration et les mêmes scénarios ont été utilisés dans le réseau WAN et le réseau LTE :

- La topologie comprend trois cellules avec une portée radio de 2000 mètres de diamètre pour chacune des cellules. ;
- dans chaque Cellule il existe un eNodeB, un noeud processeur-NDN et 25 stations mobiles (MS) qui supportent LTE ;
- toutes les MS demandent un contenu vidéo du même serveur, suivant une distribution de puissance. Par exemple, la distribution de Pareto [22] : 20 MS (80% de trafic) demandent des contenus vidéo populaires tandis que les 5 autres MS (20% de trafic) demandent des éléments vidéo impopulaires.

En ce qui concernant les politiques de gestion de cache, deux scénarios ont été proposés pour cette topologie :

- Le premier scénario : consiste à utiliser trois différentes politiques de remplacement, comme suit : LRU dans le eNodeB-1 de la cellule-1, LFU dans le eNodeB-2 de la cellule-2, et en dernier PT-LRU dans le eNodeB-3 de la cellule-3 ;
- le deuxième scénario : implémente la politique PT-Sharing dans les trois cellules, pour permettre à ces dernières de gérer leurs caches d'une manière coopérative.

### 4.6.3 Discussion

#### 4.6.3.1 Avantage

- L'utilisation d'un eNodeB réduit le problème de redondance de données et de la saturation du serveur, car il lui suffit de recevoir un seul contenu du serveur pour répondre à tous les véhicules, alors que dans le schéma sans NDN, le serveur doit envoyer la même copie de contenu à tous les véhicules ;
- réduire le délai et augmente la vitesse de transfert de données.

#### 4.6.3.2 Inconvénients

- Une grande consommation d'énergie. ;
- besoin de plus de matériel ;
- difficile à mettre en œuvre.

## 4.7 Conclusion

Le mécanisme NDN peut apporter des avantages substantiels à la communication entre véhicules dans les réseaux VANET. Car en exploitant les potentialités de ce mécanisme qui consiste à l'utilisation des noms de données, les communications peuvent commencer spontanément, sans avoir recours à d'autres procédures pour affecter les adresses IP aux différentes entités communicantes.

En outre, Les communications peuvent aussi résister aux perturbations de connectivité, grâce aux data-mules qui transportent les données dans leurs caches, ce qui augmente la probabilité de recevoir les données requises dans un délai réduit.

Le mécanisme NDN permet aussi aux véhicules de gérer les données en fonction des besoins des applications qu'ils utilisent, notamment pour les données qui dépendent de la position géographique (i.e. l'état du trafic dans un segment de route), puisque les données seront directement transmises aux consommateurs sans avoir besoin de les transférer d'abord à des serveurs distants.

Dans ce chapitre, nous avons présenté et discuter quatre différents designs VNDN. Le tableau comparatif suivant montre les différences entre les quatre architectures VNDN vues précédemment :



---

Dans le chapitre suivant nous allons présenter et évaluer les stratégies de transmissions de NDN en les adaptant à des scénarios véhiculaires.

## Chapitre 5

# Résultats et discussion

### 5.1 Introduction

Le changement radical qu'a été apporté par l'architecture NDN au paradigme de communication sur Internet, nécessite bien sûr des évaluations approfondies et multidimensionnelles des divers aspects de cette nouvelle architecture. Bien que la mise en œuvre existante de NDN, offrent des opportunités précieuses pour évaluer à la fois la conception des infrastructures et ses applications dans l'environnement réel, il reste difficile voire impossible d'expérimenter les différentes options de conception ainsi que d'évaluer leur déploiement à grande échelle.

La simulation est devenue aujourd'hui un outil primordial dans le domaine d'apprentissage et de développement des réseaux informatiques. Car elle permet de modéliser le comportement de chaque élément de configuration réseau (routeurs, Switch, etc.) ainsi que de rendre observable les principaux concepts associés aux réseaux locaux : topologie, domaines de collision et de diffusion, commutation, routage, filtrage, etc.).

Ce chapitre sera entièrement consacré à l'évaluation de performance des différentes stratégies de transmission NDN. Nous commençons tout d'abord par la présentation de l'environnement de simulation ndnSIM qui est basé sur le simulateur ns3, puis dans les sections qui suivent, nous détaillons les stratégies de transmission NDN en analysant leurs performances.

### 5.2 Le simulateur Ns3

Ns-3 [23] est un simulateur de réseau libre sous licence GNU GPLv2, destiné principalement à la recherche et à l'utilisation pédagogique. Il est considéré par beaucoup de spécialistes des télécommunications comme le meilleur simulateur à événements discrets[ref], car il est aligné sur les besoins de simulation de la recherche en réseau moderne et permettant l'ajout rapide des modèles correspondants aux technologies émergentes.

Le noyau de simulation ns-3 prend en charge la recherche sur les réseaux IP et non IP. Cependant, la grande majorité de ses utilisateurs se concentre sur les simulations sans

fil basées sur IP qui impliquent des modèles pour Wifi, WiMAX ou LTE, et une variété de protocoles de routage statiques ou dynamiques tels que OLSR et AODV. Il prend également en charge la simulation en temps réel, ce qui facilite un certain nombre de cas d'utilisation qui interagissent avec des systèmes réels. Par exemple, les utilisateurs peuvent émettre et recevoir des paquets générés sur des périphériques réseau réels, et ns-3 peut servir de cadre d'interconnexion pour ajouter des effets de liaison entre les machines virtuelles.

Ns-3 est entièrement écrite en C++ avec des bindings Python en option, ce qui permet l'écriture de scripts de simulation en C++ ou Python.

## 5.3 L'environnement de simulation ndnSIM

Pour répondre aux besoins des chercheurs et fournir à la communauté en général une plate-forme expérimentale commune, les chercheurs ont développé un simulateur NDN open source, appelé ndnSIM [24], qui fonctionne sous le simulateur de réseau NS-3.

La conception de ndnSIM a pour objectifs de permettre à la communauté de recherche de simuler toutes les opérations de base du protocole NDN sur une plate-forme de simulation open source compatible avec l'implémentation CCN (Content Centric Network), pour pouvoir partager les traces du trafic et les outils d'analyse des paquets entre CCN et ndnSIM. Il supporte également les expériences de simulation à grande échelle (routage, mise en cache, transfert de paquets, gestion de la congestion etc.).

Le simulateur ndnSIM est implémenté comme un nouveau modèle de protocoles de couche réseau, qui peut s'exécuter sur toute couche de liaison (point à point, CSMA, sans fil, etc.), couche réseau (IPv4, IPv6) et adapté à la couche de transport (TCP, UDP). Cette flexibilité permet à ndnSIM de simuler de divers scénarios de déploiement homogènes et hétérogènes (i.e. :, NDN uniquement, NDN sur IP, etc.).

En effet, le simulateur ndnSIM est implémenté de façon modulaire, en utilisant le langage C++ pour modéliser le comportement de chaque entité de la couche réseau dans NDN, telles que : CS, PIT, FIB, les interfaces réseau, les interfaces d'application, les stratégies d'acheminement des paquets, etc. Cette structure modulaire permet de modifier ou remplacer facilement n'importe quel composant avec un impact minimal sur les autres composants.

### 5.3.1 La conception de ndnSIM

La conception de ndnSIM [25] suit le principe des simulateurs de réseau NS-3, qui élaborent l'abstraction maximale pour tous les composants modélisés. De la même manière que les piles IPv4 et IPv6 existantes, ce nouveau simulateur a été conçu comme une pile

de protocoles indépendante qui peut être installée sur un nœud de réseau. Il comprend en outre un certain nombre d'applications appelées « générateur de trafic », et des classes d'assistance pour simplifier la création des scénarios de simulation, ainsi que des outils de statistiques pour rassembler et évaluer les résultats. La [figure 5.1] résume les composants abstraits implémentés dans ndnSIM :

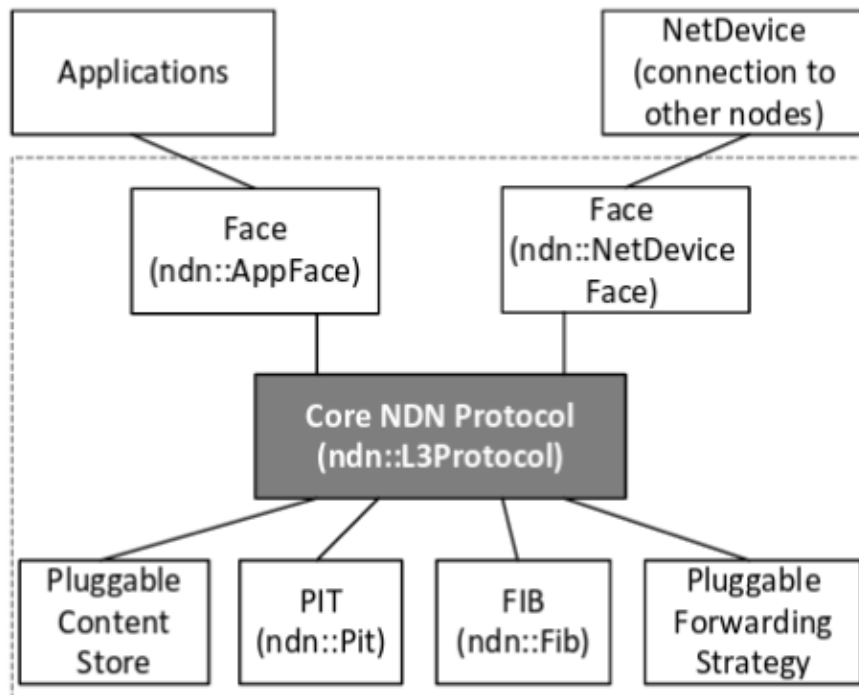


FIGURE 5.1 – Les composants de ndnSIM [25].

### 5.3.1.1 ndn : :L3Protocol

C'est une entité architecturale qui implémente le noyau NDN et ses interactions, elle permet de recevoir les IntPK et les DataPK à partir des couches supérieures et inférieures à travers ndn : :Face. Autrement dit, l'entité ndn : :L3Protocol est un agrégateur de composants logiques de toutes les communications disponibles avec les applications et les nœuds du réseau.

### 5.3.1.2 ndn : :Face

C'est l'abstraction d'une interface qui permet la communication avec les applications (ndn :AppFace) et les nœuds du réseau simulé (ndn : :NetDeviceFace). Elle supporte aussi les modules d'atténuation de la congestion au niveau de la couche liaison.

Toute communication entre ndn : :L3Protocol, le réseau et les applications s'effectue grâce à l'entité ndn : :Face , qui peut être réalisée sous diverses formes [voir Figure 5.2] :

- Link Layer Face (`Ndn::NetDeviceFace`) pour la communication inter-nœud qui est directement effectuée sur la couche liaison ;
- Network Layer Face (`ndn::Ipv4Face` et `ndn::Ipv6Face`) et Transport Layer Face (`ndn::TcpFace` et `ndn::UdpFace`) pour la communication entre les couches intermédiaires d'un nœud ;
- Application Layer Face (`ndn::AppFace`) pour les communications intra-nœud.

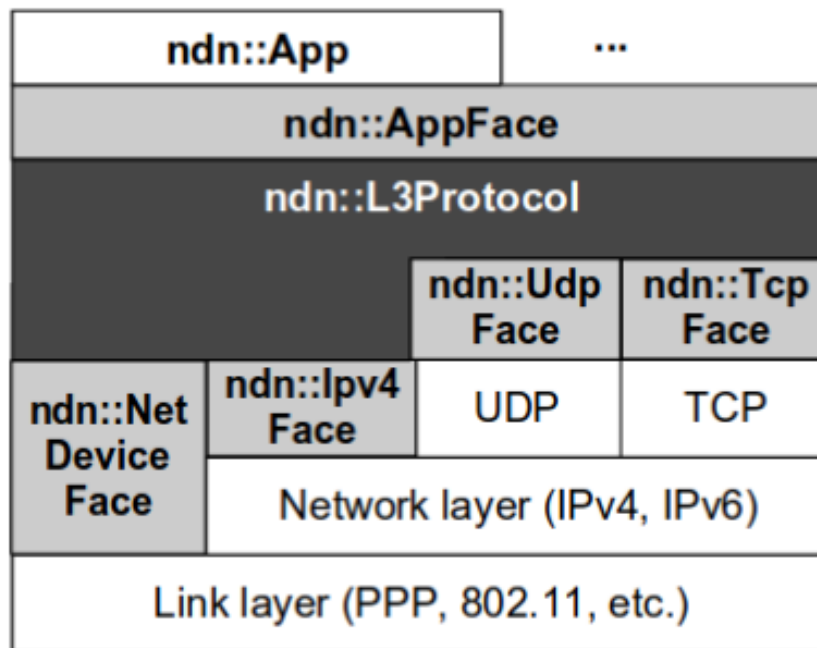


FIGURE 5.2 – Abstraction de couche de communication dans les scénarios ndnSIM [25].

### 5.3.1.3 `ndn::ContentStore`

Représente une abstraction du content store (CS) qui permet le stockage des DataPK dans le réseau (le stockage peut être à court terme, à long terme, ou permanent).

### 5.3.1.4 `ndn::PIT`

Représente une abstraction de la pending Interest table (PIT) qui conserve la trace (par préfixe) des interfaces à partir desquels les IntPK ont été reçus, et les interfaces auxquels les IntPK ont été transmis.

### 5.3.1.5 `ndn::FIB`

Représente une abstraction de la forwarding information base (FIB), qui peut être utilisé pour guider le transfert des IntPK par la stratégie d'acheminement.

### 5.3.1.6 ndn : :ForwardingStrategy

Représente l'abstraction et l'implémentation du noyau de l'acheminement des paquets IntPK et DataPK. Chaque étape dans le processus d'acheminement dans ndn : :ForwardingStrategy y compris la recherche dans le CS, PIT, FIB et l'acheminement des DataPK suivant les entrées PIT, est représenté comme un appel virtuel de fonction, qui peut être modifié dans des classes d'implémentation de stratégies d'acheminement particulières.

Chaque composant à l'exception du noyau ndn : :L3Protocol a un certain nombre d'implémentations alternatives qui peuvent être arbitrairement choisies par le scénario de simulation à l'aide de classes auxiliaires. Par exemple, ndnSIM fournit actuellement des implémentations pour l'abstraction de CS avec les règles Least-Recently-Used (LRU), First-In-First-Out (FIFO).

## 5.4 Simulation et évaluations de la performance des stratégies de transmission

La version actuelle de ndnSIM fournit les 4 stratégies suivantes, qu'on va évaluer et comparer leurs performances dans la suite de ce chapitre :

### 5.4.1 La stratégie Best-Route

La stratégie Best-Route [26] a comme caractéristique :

- Envoyer les paquets IntPK vers le upstream qui a le plus faible coût de routage (sauf le downstream) ;
- après cela, si le consommateur retransmet l'IntPK (avec un nouveau nonce) ou un IntPK similaire arrive d'un autre downstream, l'ancien IntPK est supprimé s'il s'agit d'un MIN-RETRANSMISSION-INTERVAL, sinon il est retransmis à nouveau ;
- MIN-RETRANSMISSION-INTERVAL est actuellement fixé à une valeur de 100 millisecondes ;
- un IntPK retransmis est renvoyé vers le upstream qui a le plus faible coût de routage et qui n'a pas été utilisé auparavant. Si tous les upstreams ont été utilisés, le IntPK est transmis vers le tout premier upstream utilisé ;
- la stratégie Best-Route est mise en œuvre sous la forme d'une classe nfd : : BestRouteStrategy2 classe.

### 5.4.2 La stratégie de contrôle des clients (Client Control Strategy)

Cette stratégie [26] permet aux applications locales du consommateur de choisir l'interface sortante (Outgoing-Face ) de chaque IntPK :

- Lorsqu'un LocalFace reçoit un IntPK contenant un champ NextHopFaceId dans l'entête LocalControlHeader, et que ce LocalFace autorise la fonctionnalité NextHopFaceId, alors le paquet IntPK sera transmis vers l'interface sortante spécifiée dans le champ NextHopFaceId si cette interface existe, ou rejeté dans le cas inverse ;
- si le consommateur laisse le champs NextHopFaceId vide alors le IntPK est transmis à la même manière de la stratégie Best-Route ;
- cette stratégie est implémentée sous forme d'une classe, `nfd :: ClientControlStrategy`.

### 5.4.3 La stratégie NCC

C'est une implémentation de la stratégie par défaut de CCNx 0.7.2[26]. Cette stratégie est mise en œuvre sous la forme d'une classe `nfd :: NccStrategy`.

### 5.4.4 La stratégie Multicast

La stratégie de Multicast [26]transmet tous les IntPK vers tous les upstreams, indiqués par les entrées FIB fournies.

### 5.4.5 Paramètres de simulation

Pour évaluer les performances des techniques de forwarding dans les réseaux VANET, nous avons généré le modèle de mobilité urbain avec le générateur de mobilité VanetMobiSim. Les autres paramètres de simulation sont :

TABLE 5.1 – Paramètres de simulation

Paramètre	Valeur
La surface de simulation	900x900 m
La portée de communication des nœuds	50 m
Nombre de nœuds	[50, 100, 150, 200, 250, 300]
Nombre de requêtes par seconde	[2, 3, 5, 7]
Nombre de producteurs	[5, 10, 20, 50, 70]
Le temps de simulation	300 s

## 5.4.6 Les métriques de performance

Pour comparer les stratégies de transmission de NDN, nous devons d'abord énumérer les métriques que nous allons utiliser dans cette comparaison.

### 5.4.6.1 Le taux de délivrance des paquets (Packets Delivery Ratio - PDR)

C'est le rapport entre le nombre de paquets correctement reçus (par toutes les destinations du trafic) et le nombre de paquets émis (par toutes les sources de trafic). Cette métrique est comprise dans l'intervalle  $[0, 1]$ , elle représente la fiabilité de la stratégie de transmission pour expédier tous les paquets de données envoyés. Plus le PDR converge vers 1, plus la stratégie de transmission est efficace.

### 5.4.6.2 Le temps de livraison des données (data delivery time)

Le temps de livraison des données représente le temps écoulé entre le moment d'envoi du premier IntPK par le nœud source et le moment de réception du DataPK correspondant par ce nœud. Il inclut le temps de latence pour la découverte de routes, le temps de passage des IntPK dans les PIT des nœuds intermédiaires, le temps de transmission d'un saut vers un autre, et le temps de retransmission des paquets.

### 5.4.6.3 Évaluation du taux de délivrance des paquets (PDR)

Dans l'environnement urbain, le réseau routier est dense, la vitesse des véhicules est basse et l'espace entre les routes comprend des immeubles, arbres, panneaux de circulation et des obstacles variés. A cause de ces obstacles, les communications sont limitées au réseau routier du fait que le signal sans fil a des difficultés à les traverser cela explique pourquoi on a obtenu des taux de délivrance de paquets (PDR) inférieure à 60%.

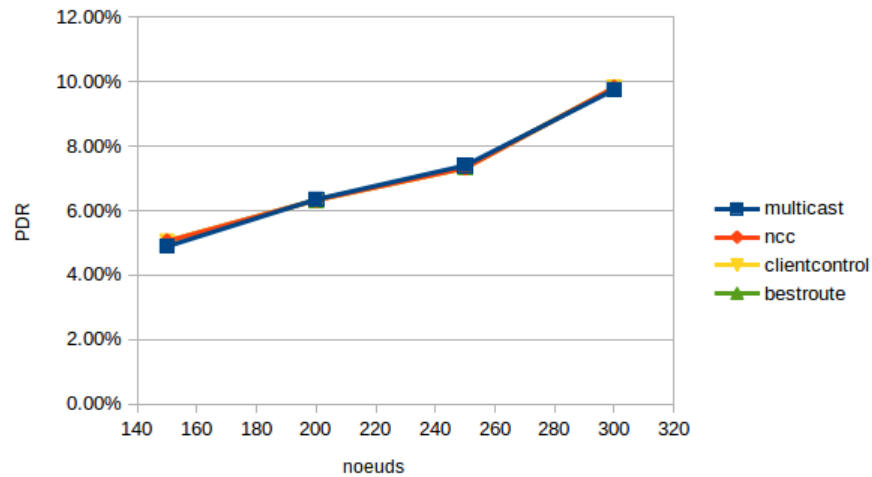


FIGURE 5.3 – Le PDR calculé en fonction du nombre de noeuds.

La [figure 5.3] ] indique le PDR calculé en fonction du nombre de noeuds. On remarque que les courbes sont pratiquement identiques sauf que, pour celle de Multicast qui diffère légèrement (d'une différence de 0,1%). Le graphe montre qu'au delà d'une densité de 150 noeuds Le PDR croît de façon presque linéaire de 5% jusqu'à 9,8% (9,7% pour Multicast) à cause de l'augmentation des noeuds producteurs. En effet, avec un nombre élevé des producteurs augmente la probabilité pour un requester, de trouver au voisinage, un noeud ayant la donnée demandée, ce qui augmente le taux de délivrance. Cependant, on remarque que ce taux reste très faible puisque les stratégies de forwarding ne sont pas adaptées aux réseaux à forte mobilité, ce qui montre la nécessité de les adapter, ou de proposer d'autres techniques pour les réseaux VANets.

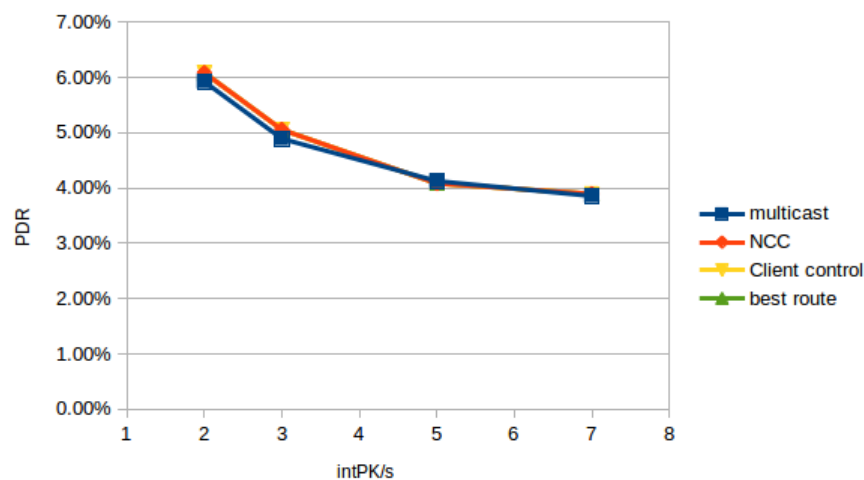


FIGURE 5.4 – Le PDR calculé en fonction du nombre IntPK.

La [figure 5.4] montre le PDR calculé en fonction du nombre des IntPK envoyés par seconde. On remarque que le PDR diminue de façon assez constante à l'augmentation du nombre de intPKs générés par nœud et par seconde, il est passé de 6% (5.9% pour Multicast) à 3.8%. Cela est dû à la croissance du nombre de collisions entre les intPKs. Après l'augmentation à chaque étape de la fréquence de génération.

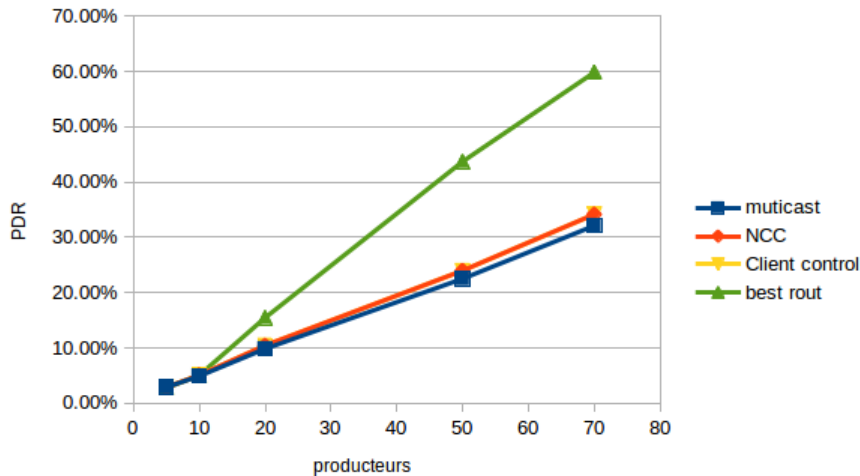


FIGURE 5.5 – Le PDR calculé en fonction du nombre de producteurs.

De la figure [figure 5.5] montrant le taux de délivrance de paquet (PDR) en fonction du nombre de producteurs, on remarque que le PDR croît à la croissance du nombre de producteurs, c'est à dire que plus le nombre de producteurs est élevé plus la probabilité de recevoir un DataPK augmente. En effet le résultat de simulation est pratiquement le même pour les quatre stratégies de transmission sauf que, pour la stratégie de Multicast, on marque une légère différence (0,1%) par rapport aux autres stratégies.

#### 5.4.6.4 Évaluation du temps de livraison des données

La [figure 5.6] montre le temps de livraison des données calculé en fonction du nombre des nœuds :

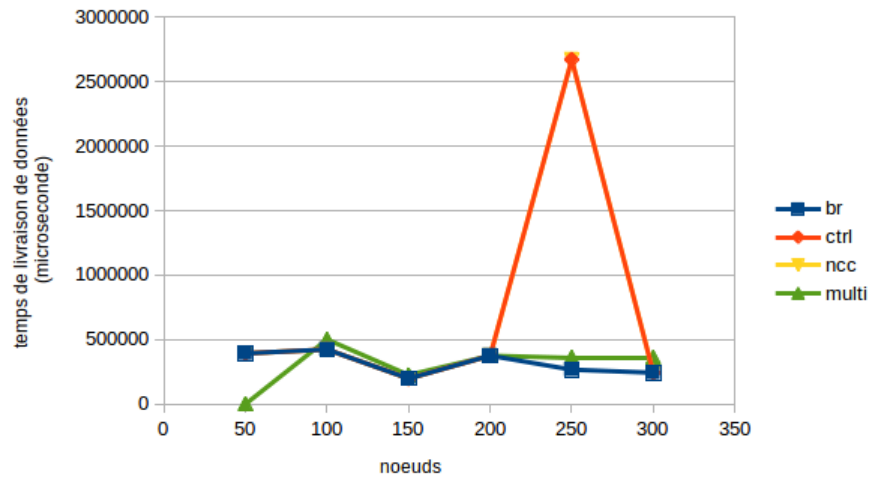


FIGURE 5.6 – Le temps de livraison de données calculé en fonction du nombre des nœuds.

Le résultat montré dans ce graphe est dû au modèle de mobilité qu'on a adopté. Le pic de la courbe Client-Control s'explique par l'isolation des nœuds. En effet, la stratégie Client-Control permet aux applications locales du nœud consommateur de choisir l'interface sortante (Outgoing-Face) de chaque IntPK. Dans ce cas, si le nœud est isolé, ses requêtes ne pourraient pas atteindre un nœud producteur convenable, ce qui augmente le temps de délivrance de données.

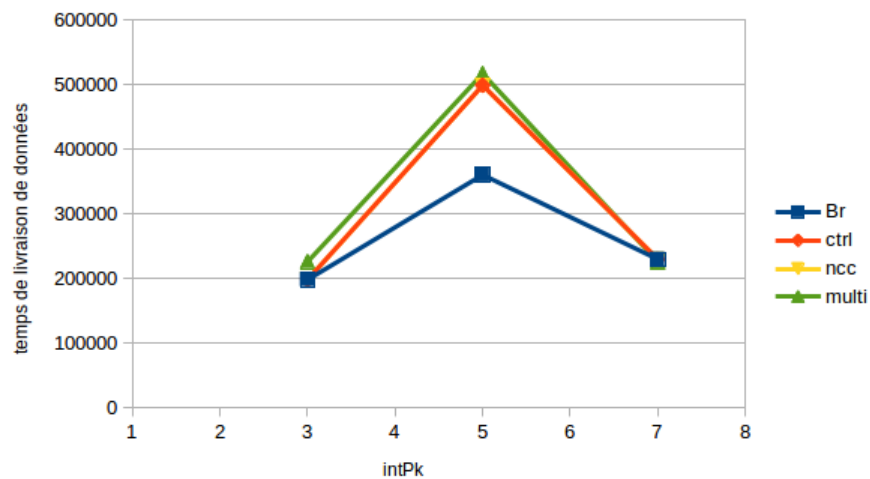


FIGURE 5.7 – Le temps de livraison de données calculé en fonction du nombre des IntPK (en microseconde).

En analysant le graphe de la [figure 5.7] qui montre le temps de livraison de données

calculé en fonction du nombre des IntPK. On remarque que les courbes croissent linéairement au début jusqu'à atteindre 0.5 s pour la stratégie multicast, 0.49 s pour NCC et client control, et 0.35 s. Cela s'explique par la collision qui se produit en augmentant la fréquence d'envoi des IntPKs. Mais, par la suite le temps de délivrance se diminue jusqu'à 0.23 s à cause du caching, qui consiste à sauvegarder par un nœud tous les paquets de données qui le traversent, ce qui permet de satisfaire les demandes « intPK rapidement après un saut généralement.

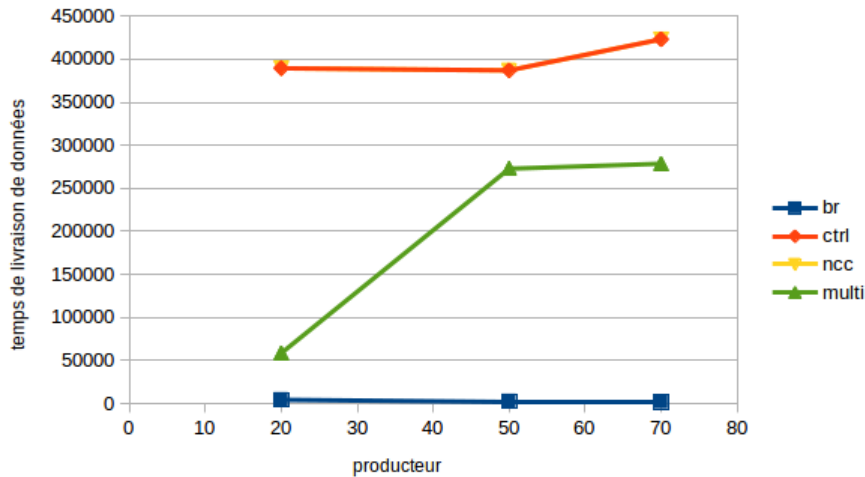


FIGURE 5.8 – Le temps de livraison de données calculé le nombre des producteurs (en microseconde).

La [figure 5.8] montre le temps de livraison de données en fonction du nombre de producteurs dans le réseau. En analysant le graphe on trouve que les courbes des stratégies Multicast ; NCC ; et Client Control sont croissantes, tandis que la courbe de la stratégie Best Route est décroissante, cette diminution de temps de délivrance de données est due à la caractéristique de la stratégie Best Route qui consiste à choisir le meilleur chemin vers le producteur.

## 5.5 Conclusion

Dans ce chapitre, nous avons évalué les performances des quatre stratégies de transmission de NDN : NCC ; Best-Route ; Client-Control ; Multicast, en adoptant un scénario de mobilité au sein d'une zone urbaine. D'après les résultats de simulation présentés précédemment on peut conclure que les stratégies de transmission de NDN ne permettent pas de surmonter les problèmes de VANET dans les zones urbaines en raison du nombre élevé de collisions et des obstacles qui caractérisent ces environnements.

## Chapitre 6

# Conclusion générale

Au cours des dernières années, les réseaux véhiculaires ad hoc (VANET) ont intéressé les constructeurs d'automobiles, les chercheurs universitaires, les agences gouvernementales ainsi que d'autres organismes de normalisation. Par conséquent, de nombreux standards, applications et mécanismes de routage ont été proposés pour répondre aux spécificités de cette classe de réseaux. En effet, cette attention croissante aux réseaux VANET, est due principalement à leur impact social et économique profond sur le plan mondial, qui est lié notamment à la grande gamme d'applications conçues pour ces environnements véhiculaires, telles que les applications de sécurité et d'efficacité du transport, qui sont employées pour diminuer le risque d'accidents de voitures (surtout dans les zones urbaines), et pour optimiser également l'échange de flux entre véhicules, ainsi que de fournir des services de confort et de commodité aux voyageurs.

Malgré leur grand potentiel et impact sur la vie quotidienne, les VANET ont en revanche de nombreux défis techniques qui découlent principalement de la conception TCP/IP qui se retrouve aujourd'hui incapable de gérer efficacement cet environnement fortement dynamique surtout sous l'effet de la diversité spatio-temporelle, la densité du trafic et de la propagation des ondes radio en environnement extérieur défavorable à l'établissement des communications sans fil.

Devant ces défis techniques, il devient indispensable de trouver un nouveau paradigme dans les réseaux de véhicules qui peut être un remplaçant ou même un compagnon de l'architecture TCP/IP. Le paradigme Information Centric Networking (ICN) a été préconisé par un certain nombre de groupe de recherche qui travaillent sur les futures architectures d'Internet.

Named Data Networking (NDN) est une architecture nouvellement proposée pour le futur Internet, issue du projet antérieur Content Centric Networking (CCN), et représente une instance pionnière d'ICN. Elle souligne une nouvelle direction pour concevoir des réseaux VANET. En se concentrant sur la récupération des données souhaitées plutôt que sur l'établissement des sessions entre les deux bouts de communication. Elle élimine également l'utilisation des adresses IP des nœuds en récupérant les données en utilisant directement les noms de données spécifiés par les consommateurs.

En effet, NDN peut fournir plusieurs avantages aux réseaux VANET. Sa sémantique axée sur les données couplée du mécanisme de mise en cache permet de garantir une fiabilité et robustesse de transmission, même sous les pires conditions environnementales (forte mobilité, connectivité intermittente, etc.).

Dans ce mémoire nous avons présenté et discuté les caractéristiques, les avantages et les inconvénients de quatre designs VNDN qui sont le résultat de l'intégration de NDN aux VANET traditionnels. Le premier design appelé VNDN permet à un véhicule de jouer 4 rôles en même temps : forwarder, producer, consumer, et Data-mule. Les auteurs de VNDN ont proposé aussi une amélioration de la diffusion WIFI pour les communications V2V.

Le deuxième design VNDN est appelé Navigo, il propose une nouvelle approche permettant de relever les défis des VANET, tels que les perturbations fréquentes de la connectivité et les changements brusques de la topologie du réseau véhiculaire. Le troisième design appelé HVNDN est un prototype d'implémentation d'une architecture hiérarchique permettant la distribution de contenu dans le réseau de véhicules. Le quatrième design appelé VEDNET qui supporte trois mécanismes différents de communication de véhicules, (V2I), (V2R), et (V2V).

Nous avons ainsi présenté et évalué les stratégies de transmission de NDN : Best-Route, NCC, Multicast, et client control dans les réseaux NDN tout en adoptant des scénarios VANET dans un environnement urbain. D'après les résultats de simulation présentés précédemment on peut conclure que les stratégies de transmission de NDN ne permettent pas de surmonter les problèmes de VANET dans les zones urbaines en raison du nombre élevé de collisions et des obstacles qui caractérisent ces environnements.

## 6.1 Perspectives

Dans ce présent travail nous avons simulé les stratégies de transmission NDN en adoptant un modèle de mobilité VANET au sein d'une zone urbaine. Comme travaux futures, nous proposons d'implémenter les stratégies de transmission de NDN dans un réseau purement VNDN.

Nous proposons aussi de combiner les avantages de chacune des architectures VNDN discutées précédemment, pour aboutir à une nouvelle architecture VNDN performante, permettant ainsi de faire face aux défis actuels des VANET.

# Bibliographie

- [1] URL : <https://named-data.net/>.
- [2] M.JERBI. “Protocoles pour les communications dans les reseaux de vehicules en environnement urbain”. 11, 2008.
- [3] NADIA HADDADO. “Réseaux ad hoc vehiculaire : vers une dessimination de données efficace coopérative et fiable”.
- [4] Vishal KUMAR<sup>1</sup>, Shailendra MISHRA<sup>1</sup> et Narottam CHAND<sup>2</sup>. “Applications of VANETs : Present and Future”.
- [5] Jonathan PETIT. “Surcoût de l’authentification et du consensus”.
- [6] Shereen A. M. AHMED<sup>1</sup>, Sharifah H. S. ARIFFIN<sup>2</sup> et Norsheila FISAL<sup>3</sup>. “Overview of Wireless Access in Vehicular Environment (WAVE) Protocols and Standards”.
- [7] “Intelligent Transport Systems (ITS); Communications Architecture”. In : *ETSI* (sept. 2010).
- [8] Andrew CHEN et al. “Smoothing Vehicular Traffic Flow using Vehicular-based Ad Hoc Networking and Computing Grid (VGrid)”.
- [9] SALIM BITAM, ABDELHAMID MELLOUK et SHERALI ZEADALLY. “VANET-CLOUD : A GENERIC CLOUD COMPUTING MODEL FOR VEHICULAR AD HOC NETWORKS”.
- [10] MAHSIMA RAHIMI. “CLOUD BASED IP MULTIMEDIA SUBSYSTEM (IMS) ARCHITECTURE TO INTEGRATE VEHICULAR AD HOC NETWORK (VANET) AND IMS”.
- [11] Asad Vakili HADIS REZAEI. “Named Data Networking : Investigate of New Infrastructure for Future of Internet”.
- [12] “Traduction adaptée par Jean-Pierre Kuypers du RFC 1462 FYI on "What is the Internet ?"”. In : (mai 1993).
- [13] Marica AMADEO et al. “Content-centric wireless networking : A survey”.
- [14] Wentao SHANG et al. “Named Data Networking of Things (Invited Paper)”.
- [15] Marica Amadeo Claudia CAMPOLO et Antonella MOLINARO. “CRoWN : Content-Centric Networking in Vehicular Ad Hoc Networks”.

- 
- [16] Giulio Grassi Davide Pesavento Giovanni Pau Rama Vuyyuru Ryuji Wakikawa Lixia ZHANG. “VANET via Named Data Networking”.
- [17] Giovanni PAU. “VNDN : Vehicular Named Data Networks”.
- [18] Giulio Grassi Davide Pesavento Giovanni Pau Lixia Zhang Serge FDIDA. “Navigo : Interest Forwarding by Geolocations in Vehicular Named Data Networking”.
- [19] G. Grassi 1 D. Pesavento G. Pau L. Zhang S. FDIDA. “Navigo : Interest Forwarding by Geolocations in Vehicular Named Data Networking”.
- [20] Cristina De Castro-Carla Raffaelli-Oreste ANDRISANO. “A Dynamic Hierarchical VANET Architecture for Named Data Networking Applications”.
- [21] DungOngMaua YinZhanga TarikTalebb Victor C.M.LEUNG. “VENDNET : Vehicular Named Data NETWORK MinChena”.
- [22] URL : [https://fr.wikipedia.org/wiki/Principe\\_de\\_Pareto](https://fr.wikipedia.org/wiki/Principe_de_Pareto).
- [23] URL : <https://www.nsnam.org/>.
- [24] URL : <http://ndnsim.net/>.
- [25] Alexander Afanasyev Ilya MOISEENKO et Lixia ZHANG. “ndnSIM : NDN simulator for NS-3”.
- [26] Alexander Afanasyev Junxiao Shi Beichuan Zhang Lixia Zhang Ilya Moiseenko Yingdi Yu Wentao Shang Yi Huang Jerald Paul Abraham Steve DiBenedetto Chengyu Fan Christos Papadopoulos Davide Pesavento Giulio Grassi Giovanni Pau Hang Zhang Tian Song Haowei Yuan Hila Ben Abraham Patrick Crowley Syed Obaid Amin Vince LEHMAN et Lan WANG. “NFD Developer’s Guide”.