HAL
open science

Identification de l'environnement basée sur l'estimation de canal et génération de clés de sécurité pour les communications véhiculaires

Soheyb Ribouh

To cite this version:

Soheyb Ribouh. Identification de l'environnement basée sur l'estimation de canal et génération de clés de sécurité pour les communications véhiculaires. Electronique. Université Polytechnique Hauts-de-France, 2020. Français. NNT : 2020UPHF0029 . tel-03204589

HAL Id: tel-03204589

https://theses.hal.science/tel-03204589

Submitted on 21 Apr 2021

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Thèse de doctorat

Pour obtenir le grade de Docteur de

l'UNIVERSITÉ POLYTECHNIQUE HAUTS-DE-FRANCE
et l'INSA HAUTS-DE-FRANCE

Discipline : **Télécommunication**

Présentée et soutenue par : Soheyb RIBOUH.
Le 18/12/2020, à Valenciennes.

École doctorale	:	Sciences Pour l'Ingénieur (ED SPI 072).
Laboratoire	:	IEMN-DOAE UMR CNRS 8201.

IDENTIFICATION DE L'ENVIRONNEMENT BASÉE SUR L'ESTIMATION DE CANAL PAR DEEP LEARNING ET GÉNÉRATION DE CLÉS DE SÉCURITÉ POUR LES COMMUNICATIONS VÉHICULAIRES

JURY

Président du jury	:	Mr. Toufik AHMED	Professeur à l'Université de Bordeaux, INP. France
Rapporteurs	:	Mr. Vahid MEGHDADI	Professeur à l'Université de Limoges, ENSIL. France
		Mme. Maria-Gabriella DI BENEDETTO	Professeur à l'Université La Sapienza, Rome. Italie
Examinateurs	:	Mr. Abdenour HADID	Professeur à l'Université de Oulu, Finland .
		Mr. Mohammad Abdullah AL FARUQUE	Professeur à l'Université d'Irvine Californie, UCI. USA .
Directrice de thèse	:	Mme. Atika RIVENQ	Professeur à l'Université Polytechnique Hauts-de-France, UPHF. France
Encadrant de thèse	:	Mr.Yassin ELHILLALI	MCF-HDR à l'Université Polytechnique Hauts-de-France, UPHF. France.

2020

Résumé

Les véhicules connectés autonomes devraient être largement déployés dans le cadre de la prochaine génération de systèmes de transport et de la future route intelligente. Comme l'environnement véhiculaire est très mobile, les messages transmis sont affectés par l'effet de canal sans fil. L'estimation de canal devient ainsi l'une des tâches les plus difficiles des communications véhiculaires (V2X). Dans ce contexte, nous proposons un nouvel algorithme d'estimation de canal MSCR (Multiple Sequential Constraint Removal) dédié aux communications V2X basées sur les systèmes OFDM. De plus, l'identification de l'environnement dans lequel le véhicule circule est assez importante pour permettre au véhicule de prendre les bonnes décisions de conduite autonome. Ainsi, en exploitant les caractéristiques estimées du canal sans fil, nous proposons une nouvelle approche d'identification de l'environnement véhiculaire basée sur le Deep Learning, où les CSI estimées sont utilisées comme caractéristiques d'entrée pour le modèle proposé. En outre, comme l'environnement véhiculaire est un accès ouvert, les messages échangés via la communication V2X sont vulnérables aux attaques. Par conséquent, nous proposons un nouvel algorithme de génération de clé à base de la couche physique afin de sécuriser les communications véhiculaires, où les valeurs CSI sont utilisées comme source aléatoire. Notre algorithme de génération de clés a été implémenté sur des cartes SDR USRP et a été testé sur un environnement véhiculaire réel. Les résultats de l'évaluation des performances de toutes les méthodes proposées (algorithme d'estimation de canal MSCR, approche d'identification d'environnement et algorithme de génération de clés), montrent qu'elles répondent parfaitement aux exigences requises dans un environnement véhiculaire.

Mots clés : Communications V2X, Estimation du canal, Deep learning, Génération de clé à base de la couche physique, Sécurité V2X, Identification de l'environnement véhiculaire, ITS-G5.

Abstract

Autonomous connected vehicles are expected to see prevalent usage as part of the next generation of transportation systems and the smart road vision Since the vehicular environment is highly mobile, the transmitted messages are affected by the wireless channel effect. This makes channel estimation one of the challenging tasks in Vehicle-To-Everything (V2X) communications. In this context, we propose a novel Multiple Sequential Constraint Removal (MSCR) algorithm of channel estimation, dedicated to OFDM systems based-V2X communications. In addition, identifying the environment where the vehicle is driving along is quite important in order to allow the vehicle to make the correct self- driving decisions. Thus by exploiting the estimated wireless channel characteristics, we propose a novel vehicular environment identification approach based on deep learning, where the estimated channel state information (CSI) are used as input features for the proposed model. Besides that, as the vehicular environment is open access exchanged messages through V2X communication are vulnerable to attacks. Therefore, we propose a novel physical layer key generation algorithm in order to secure vehicular communications, where the CSI values are used as a source of randomness. Our key generation algorithm has been implemented on USRP Software-Defined Radios (SDR) cards and has been tested on a real-world testbed vehicular environment. The performance evaluation results of all the proposed methods (MSCR channel estimation algorithm, environment identifications approach, and key generation algorithm), show that they meet the performances required in a vehicular environment.

Keywords : V2X communication, Channel State Information (CSI), Deep learning, Physical layer key generation, V2X security, Vehicular environment identification, ITS-G5.

À mon chère Papa, grâce à lui je suis devenue ce que je suis aujourd'hui, qui était toujours fière de moi et attendait le jour de me voir docteur, mais malheureusement il nous a quittés un mois avant la soutenance.

À ma chère Maman, qui m'a tout donné sans rien me demander, qui m'accompagnait durant mon parcours depuis mon enfance, et qui était toujours fière de moi.

À mes deux sœurs Asma et Oumaima, qui m'ont encouragé durant toute la période thèse

Je dédie ce travail.

Remerciement

Je tiens à remercier toutes les personnes qui ont contribué à ce travail.

Tout d'abord, j'adresse mes remerciements à ma directrice de thèse **Mme Atika RIVENQ** , grâce à elle ce travail a pu aboutir. Je la remercie chaleureusement pour sa présence, son soutien inconditionnel, ses qualités humaines et sa confiance qui m'ont accompagnée durant ces années.

Je tiens à remercier vivement **Professeur Mohammad Abdullah AL FARUQUE** pour son bon accueil au sein de son laboratoire de recherche durant mon stage à Irvine USA, pour sa disponibilité, sa gentillesse et le partage de son expérience.

Je tiens à remercier mon encadrant **Mr Yassin EL HILLALI** , pour son aide, le temps passé ensemble, son écoute et ses conseils.

Je présente également mes remerciements à tous les membres du jury :
Mr. Abdenour HADID, Mr. Toufik AHMED, Mme.Maria-Gabriella DI BENEDETTO, Mr. Mohammad Abdullah AL FARUQUE et Mr. Vahid MEGHDADI qui ont accepté d'examiner mon travail et pour l'intérêt qu'ils ont porté à ce dernier.
Je tiens à remercier également **Professeur fadi kurdah**, Directeur du laboratoire **CECS** à Irvive pour son bon accueil.

J'adresse mes vifs remerciements à mes parents et mes deux soeurs pour leurs accompagnements et encouragements permanents.

Je tiens à remercier tous les membres des laboratoires **IEMN** à Valenciennes et **CECS** à Irvive : collègues et amis, qui ont su m'aider, apporter leur soutien et faire de cette période de thèse une profonde réussite professionnelle et personnelle.

Table des matières

1 Introduction générale **11**
1.1 Problématique . . . 12
1.2 Challenges de recherche . . . 13
1.3 Nos contributions . . . 13
1.4 Organisation du manuscrit . . . 14

2 Systèmes de Transports Intelligents (ITS) **15**
2.1 Introduction . . . 16
2.2 L'architecture des communications dédiées aux ITS . . . 16
2.3 Les systèmes de transports intelligents coopératifs (C-ITS) . . . 18
2.4 Les communications véhiculaires (V2X) . . . 18
2.5 Projets européens dédiées aux C-ITS . . . 19
2.6 Technologies de communications véhiculaires . . . 21
2.7 Conclusion . . . 37

3 Estimation du canal véhiculaire **39**
3.1 Introduction . . . 40
3.2 Modélisation d'un système de communication véhiculaire . . . 41
3.3 Canal de propagation véhiculaire . . . 43
3.4 Techniques d'estimation de canal . . . 45
3.5 Algorithmes proposés . . . 50
3.6 Évaluation des performances . . . 52
3.7 Conclusion . . . 57

4 Identification de l'environnement véhiculaire **59**
4.1 Introduction . . . 60
4.2 Machine learning . . . 61
4.3 Deep Learning . . . 63
4.4 Méthodologie . . . 66
4.5 Évaluation des performances et résultats . . . 70
4.6 Conclusion . . . 82

5 Sécurité des communications véhiculaires **83**
5.1 Introduction . . . 84
5.2 État de l'art sur la sécurité pour les communications véhiculaires. . . . 85
5.3 Processus de génération de clés . . . 87
5.4 Algorithme proposé . . . 87
5.5 Tests expérimentaux . . . 90
5.6 Évaluation des performances de l'approche CSI . . . 98
5.7 Conclusion . . . 104

6 Conclusion générale **105**
6.1 Conclusion . . . 106
6.2 Perspectives . . . 107

Table des figures

2.1 Architecture des ITS 16
2.2 Différents types de communications V2X 18
2.3 Déploiement du projet InterCor 20
2.4 Couche d'accès d'ITS-G5 21
2.5 Structure de paquet de la couche PHY 23
2.6 Attribution des canaux pour la norme ITS-G5 [22] 25
2.7 Architecture LTE. 27
2.8 Architecture NSA 5G 30
2.9 Architecture SA 5G 31
2.10 Architecture générale 5G 31
2.11 Communication V2X via l'interface PC5 et l'interface LTE-Uu 33
2.12 Architecture globale C-V2X 33
2.13 Architecture NR-V2X -Alternative 1- 36
2.14 Architecture NR-V2X -Alternative 2- 36
2.15 Architecture NR-V2X -Alternative 3- 37

3.1 Modèle d'un système de communication véhiculaire. 41
3.2 Émetteur IEEE 802.11p 41
3.3 Récepteur IEEE 802.11p 42
3.4 Signal à trajets multiples reçu dans un environnement véhiculaire. . . 43
3.5 Canaux sélectifs en fréquence (Présence des ISI) 45
3.6 Allocation des symboles pilotes et des séquences d'apprentissages. . . . 47
3.7 Évaluation de l'algorithme MSCR avec différentes valeurs K 52
3.8 SER vs. SNR des algorithmes MSCR, SCR et des estimateurs MMSE, LS dans un environnement Rural-LOS. 53
3.9 SER vs. SNR des algorithmes MSCR, SCR et des estimateurs MMSE, LS dans un environnement Highway-LOS. 53
3.10 SER vs. SNR des algorithmes MSCR, SCR et des estimateurs MMSE, LS dans un environnement Highway-NLOS. 54
3.11 SER vs. SNR des algorithmes MSCR, SCR et des estimateurs MMSE, LS dans un environnement Urban NLOS 54
3.12 SER vs. SNR des algorithmes MSCR, SCR et des estimateurs MMSE, LS dans un environnement Urban LOS 55

4.1 neurone 63
4.2 réseaux de neurones artificiels (ANN) 64
4.3 Opération de convolution 65
4.4 Organigramme décrivant le processus d'identification de l'environnement véhiculaire 66
4.5 Architecture CNN proposée 67
4.6 Architecture ANN proposée 68
4.7 Matrice de confusion basée sur les séquences LTS pour l'architecture CNN proposée sur les échantillons de tests (Test-Set) 71

4.8 Matrice de confusion basée sur les séquences LTS pour l'architecture ANN sur les échantillons de tests (Test-Set) 71
4.9 Matrice de confusion basée sur les séquences LTS pour l'approche KNN sur les échantillons de tests (Test-Set) 73
4.10 Matrice de confusion basée sur les séquences LTS pour l'approche RF sur les échantillons de tests (Test-Set) 73
4.11 Matrice de confusion basée sur les séquences LTS pour l'approche GNB sur les échantillons de tests (Test-Set) 74
4.12 Matrice de confusion basée sur les séquences LTS pour l'approche SVM sur les échantillons de tests (Test-Set) 74
4.13 Matrice de confusion basée sur l'approche CSI pour le modèle CNN proposé 75
4.14 Matrice de confusion basée sur les valeurs CSI pour le modèle ANN . 77
4.15 Matrice de confusion basée sur les valeurs CSI pour l'approche RF . . 77
4.16 Matrice de confusion basée sur les valeurs CSI pour l'approche K-NN . 78
4.17 Matrice de confusion basée sur les valeurs CSI pour l'approche GNB . 78
4.18 Matrice de confusion basée sur les valeurs CSI pour l'approche SVM . 79

5.1 Principe de génération des clés en utilisant les caractéristiques du canal. 86
5.2 Processus de génération de clés. 87
5.3 Diagramme de constellation de la 64-QAM. 90
5.4 Configuration expérimentale avec les cartes Ettus B210 embarquées dans les véhicules. 91
5.5 Organigramme de processus de génération de clés basé sur RSSI . . . 92
5.6 Valeurs RSSI collectées des véhicules A et B dans un scénario urbain. 93
5.7 Organigramme décrivant le processus de génération de clés basé sur CSI 94
5.8 Valeurs CSI collectées des véhicules A et B dans un scénario autoroute. 95
5.9 Taux de clés similaires générées pour les véhicules A et B pour un alignement réussi et infructueux. 96
5.10 Taux de similarité pour les scénarios urbains et autoroutiers. 97
5.11 Attaque non-intrusive. 100
5.12 source d'entropie 102
5.13 Comparaison de notre min-entropie moyen avec l'état de l'art 103

Liste des tableaux

2.1 Modulation et taux de codage de la couche OFDM-PHY 22
2.2 Codage du débit 23
2.3 Bandes de fréquences ITS-G5 25
2.4 Caractéristiques techniques de la norme 802.11bd. 26
2.5 Paramètres de l'OFDM pour LTE 28
2.6 Spécifications de fréquences et de formes d'onde 5G NR Release 15 . . 29

3.1 Paramètres de l'environnement Rural LOS 44
3.2 Paramètres des environnements Urban LOS et Urban NLOS 44
3.3 Paramètres des environnements Highway LOS et Highway NLOS . . . 44
3.4 Temps d'exécution moyen des algorithmes MSCR et SCR 55
3.5 Intervalle de confiance à 15 dB de SNR 56
3.6 Intervalle de confiance à 20 dB de SNR 56

4.1 Labels des environnements véhiculaires 69
4.2 Précision de training et de test de l'approche LTS pour les configurations d'entrées : Module, Angle et 2-Canaux. 70
4.3 Comparaison de Précision de classification et de temps moyens de prédiction pour l'approche LTS 72
4.4 Précision de training et de test de l'approche CSI pour les configurations d'entrées : Module, Angle et 2-Canaux. 75
4.5 Comparaison de Précision de classification et de temps moyens de prédiction pour l'approche CSI 76
4.6 Comparaison de notre modèle CNN aux architectures de pointes . . . 81

5.1 Paramètres de génération de clés pour l'algorithme basé sur RSSI. . . 91
5.2 Clés générées à partir de mesures RSSI dans les scénarios de test urbains 94
5.3 Clés générées à partir des CSI estimées dans les scénarios autoroutiers 97
5.4 Temps d'exécution moyen pour la génération de clés. 99
5.5 Taux de discordance de bits moyen pour les clés générées par le véhicule attaquant E 100
5.6 Temps Brute Force vs longueur de clé. 101

Chapitre 1

Introduction générale

1.1 Problématique

Selon l'*Organisation Mondiale de la Santé* (OMS), chaque année, environ 1,35 million de personnes perdent leurs vies à cause des accidents de la route. Entre 20 et 50 millions de personnes supplémentaires souffrent de blessures non mortelles, et beaucoup d'entre eux souffrent d'un handicap en raison de leurs blessures.
Les accidents de la route entraînent des pertes économiques considérables pour les individus, leurs familles et les nations dans leur ensemble Les accidents de la route coûtent à la plupart des pays 3% de leur produit intérieur brut.
Les facteurs et les raisons majeurs de l'augmentation de facteur de risque d'accidents pour les usagers de la route sont :

— **L'excès de vitesse :** Une augmentation de la vitesse moyenne est directement liée à la probabilité d'un accident. Où, chaque augmentation de 1% de la vitesse moyenne produit une augmentation de 4% du risque d'accident mortel.

— **Les infrastructures routières non sécurisées :** Les routes devraient être conçues touts en assurant qu'il existe des installations adéquates pour les piétons, les cyclistes et les motocyclistes. Afin de réduire le risque d'accident chez les usagers de la route.

— **Les véhicules non sécurisés :** Les véhicules sécurisés jouent un rôle essentiel pour éviter les accidents et réduire la probabilité de blessures graves.

En plus des accidents les usagers de la route passent environ six jours et 16 heurs par année en embouteillage, selon les chiffres de la société de navigation TomTom. Selon le Texas Transportation Institute, chaque heure coincée dans la circulation coûte environ 21$ en temps et en carburant gaspillés.
En outre les causes de la congestion routière sont : les accidents,les zones de travaux, la mauvaise synchronisation des feux de circulation et les événements spéciaux.
Ces Problèmes dans le domaine du transport, liés à la congestion routière, ainsi que les conséquences relatives à l'augmentation du nombre d'accidents chaque année nécessite la mise en place d'un système de transport intelligent qui consiste à connecter les véhicules, l'infrastructure et les usagers de la route, afin de fournir plus de sécurité, plus de confort et une meilleure gestion du trafic routiers.
Les communications véhiculaires représentent une part majeure dans les systèmes de transport intelligents (ITS), qui devrait connaître une large expansion dans le cadre de la prochaine génération de systèmes de transport et la future route intelligente. Cette attente est soutenue par les tendances de l'industrie, telles que les constructeurs automobiles Volkswagen et Toyota qui ont déclaré leur intention de déployer la technologie de communication V2X, dans leurs voitures en 2021. De plus, cette initiative, est également soutenue par des tendances politiques telles que le mandat proposé par le National Highway Traffic et Administration de la sécurité (NHTSA)aux USA qui auraient exigé que tous les véhicules aient une capacité V2X d'ici début 2021. En outre, de nombreux pays européens envisagent d'installer des dispositifs de communication V2X sur les routes et les véhicules d'ici la fin de l'année prochaine.
Ces tendances elles-mêmes sont probablement motivées par les avantages promis par les ITS, qui aideraient à résoudre les problèmes actuels de mobilité routière. Les ITS ont d'ailleurs le potentiel de réduire la gravité jusqu'à 80% des collisions. En plus les données collectées à partir de ces systèmes intelligents, peuvent contribuer à une meilleure gestion de la circulation pour réduire le temps de trajet et diminuer la pollution.
D'autre part, les véhicules autonomes ont également été au centre des progrès récents des transports, qui devraient représenter 40% du trafic routier d'ici 2040. L'objectif du développement des véhicules autonomes est d'atteindre le plus haut niveau

d'autonomie (Niveau 5), là où aucun conducteur n'est requis. Lorsque cet objectif sera atteint, les communications V2X seront obligatoire pour exploiter pleinement le potentiel de ces véhicules. De plus, les communications V2X sont nécessaires pour permettre la transition de l'auto-autonomie à l'autonomie collaborative totale. En outre, un ITS pleinement réalisé composé de véhicules autonomes connectés promet d'être l'avenir des systèmes de transport.

1.2 Challenges de recherche

La mise en place d'un ITS à hautes performances, a ouvert plusieurs axes et challenges de recherche. Dans le cadre de notre travail, on propose des solutions pour résoudre les défis de recherche liés aux ITSs suivants :

— Étant donné que les véhicules connectés sont très mobiles lorsqu'ils communiquent, les signaux reçus sont fortement influencés par l'effet du canal de communication sans fil. Pour cette raison, l'estimation du canal est une tâche clé dans la communication véhiculaire.

— L'identification de l'environnement dans lequel le véhicule circule permet au véhicule de prendre une auto-décision sur la vitesse de conduite correcte. Pour cela l'intelligence artificielle s'est imposée comme un acteur de premier plan dans le développement de systèmes intelligents pour permettre aux véhicules autonomes de prendre la bonne décision en fonction de la perception de l'environnement.

— Les véhicules dans un système ITS sont très mobiles et, par conséquent, échangent des informations de manière ad hoc en formant des réseaux ad hoc de véhicules (VANET) ou des réseaux ad hoc mobiles (MANET) ; cependant, la connectivité croissante entre les véhicules et les environnements externes dans cesVANET signifie qu'il y a une augmentation des surfaces d'attaque et des vulnérabilités. Il est donc primordial de sécuriser les communications véhiculaires.

1.3 Nos contributions

Dans le cadre de ce travail de recherche et afin de faire face aux challenges décrits précédemment nous avons proposé les solutions suivantes :

1. Dans ce travail nous proposons un nouvel algorithme d'estimation de canal pour les systèmes OFDM. Cet algorithme répond aux exigences et conditions difficiles de l'environnement véhiculaires.

2. Afin qu'un véhicule autonome connecté puisse identifier l'environnement dans lequel il roule. Nous proposons une nouvelle approche d'identification de l'environnement véhiculaire basée sur l'intelligence artificielle en utilisant les messages reçus dans un réseau véhiculaire sans avoir besoin d'implémenter des capteurs tel que : caméra ou RADAR.

3. Pour sécuriser les communications véhiculaires, nous proposons une nouvelle méthode de génération de clés cryptographiques basées sur la couche physique. Nous avons introduit une nouvelle méthode de quantification qui répond aux exigences de temps de latence des communications V2X.

1.4 Organisation du manuscrit

Ce mémoire est organisé en quatre chapitres comme suit :

— Dans le premier chapitre, nous décrivons les systèmes de transports intelligents et leurs caractéristiques. notamment nous introduisons les principaux projets européens sur les ITSs. Ensuite nous décrivons les principales caractéristiques des différentes technologies de communications véhiculaires : wifi véhiculaire (IEEE 802.11p et IEEE 802.11bd) et les réseaux cellulaires (C-V2X et NR-V2X).

— Dans le deuxième chapitre, dans une première partie, nous décrivons les différents blocs qui constituent un système de communication V2X basé sur la forme d'onde OFDM. Ensuite, nous introduisons les caractéristiques d'un canal de propagation dans un environnement véhiculaire, ainsi que son effet sur les communications V2X basées sur l'OFDM. Après cela, nous indiquons les différentes classes des techniques d'estimation de canal existantes dans la littérature. Dans la deuxième partie de ce chapitre, nous proposons une nouvelle approche d'estimation de canal pour les systèmes de communication V2X basée sur l'OFDM. Ensuite nous présentons les résultats d'évaluation de performances de cette méthode dans un environnement véhiculaire.

— Dans le troisième chapitre, nous commençons tout d'abord par présenter un état de l'art sur les techniques de perception de l'environnement véhiculaire. Après nous introduisons les différents modèles de Machine Learning, et aussi les architectures de Deep Learning basées sur les réseaux de neurones utilisés dans le cadre de ce travail. Ensuite, nous décrivons notre méthode proposée pour l'identification de l'environnement, ainsi que les architectures de Deep Learning utilisées dans ce processus. Les performances de cette méthode en utilisant deux différentes approches ont été évaluées et comparés par rapport aux autres architectures existantes.

— Un état de l'art sur les solutions existantes pour sécuriser les communications véhiculaires est présenté dans le quatrième chapitre. Ensuite nous avons décrit le processus de génération de clés sur lequel nos deux algorithmes proposés sont basés. Les deux algorithmes proposés ainsi que les test expérimentaux effectués sont introduits. Une évaluation de performances en termes de Min-entropie, temps de génération de clés et de complexité est réalisée. Aussi des tests de robustesse face aux attaques non-intuitives et brute force sont présentés, et finalement, une étude comparative avec les techniques de l'état de l'art existantes est élaborée.

Chapitre 2

Systèmes de Transports Intelligents (ITS)

Sommaire

2.1 Introduction 16
2.2 L'architecture des communications dédiées aux ITS . . 16
2.3 Les systèmes de transports intelligents coopératifs (C-ITS) 18
2.4 Les communications véhiculaires (V2X) 18
2.5 Projets européens dédiées aux C-ITS 19
2.5.1 Le projet Scoop 19
2.5.2 Le projet InterCor 19
2.5.3 Le projet C-Road 19
2.5.4 Le projet InDiD 20
2.6 Technologies de communications véhiculaires 21
2.6.1 IEEE 802.11p (ITS-G5) 21
2.6.1.1 Architecture ITS-G5 21
2.6.1.2 Le paquet IEEE 802.11p 23
2.6.1.3 Fréquences occupées par ITS-G5 24
2.6.2 IEEE 802.11bd 25
2.6.3 Les réseaux cellulaires 26
2.6.3.1 Le réseau cellulaire 4G/LTE 27
2.6.3.2 La technologie 5G 28
2.6.4 Réseau cellulaire V2X (C-V2X) 32
2.6.4.1 Architecture C-V2X 33
2.6.5 NEW RADIO NR-V2X 34
2.6.5.1 Architecture NR-V2X 35
2.7 Conclusion 37

2.1 Introduction

Les systèmes de transport intelligents (Intelligent Transportation System (ITS)) sont les applications qui consistent à exploiter les technologies et les standards de l'information et de la communication dans le domaine des transports.
Les ITSs englobent généralement l'interaction dynamique entre le véhicule, son usager et la route. Ces systèmes intelligents , sont conçus afin d'assurer une meilleure sécurité routière, une réduction de la congestion routière en donnant une meilleure gestion des flux de trafic, et une conduite coopérative qui utilise efficacement la couverture routière disponible.

2.2 L'architecture des communications dédiées aux ITS

L'architecture des ITS est organisée sous forme d'une structure de plusieurs couches. Elle s'inspire du modèle de communication OSI (TCP/IP). La séparation en couches permet de développer des applications qui fonctionnent indépendamment des technologies sous-adjacentes. Elle favorise donc la portabilité sur des plates-formes matérielles et logicielles distinctes, et permet aussi de remplacer, d'ajouter des fonctionnalités ou des technologies dans les couches basses sans incidence sur les couches hautes.

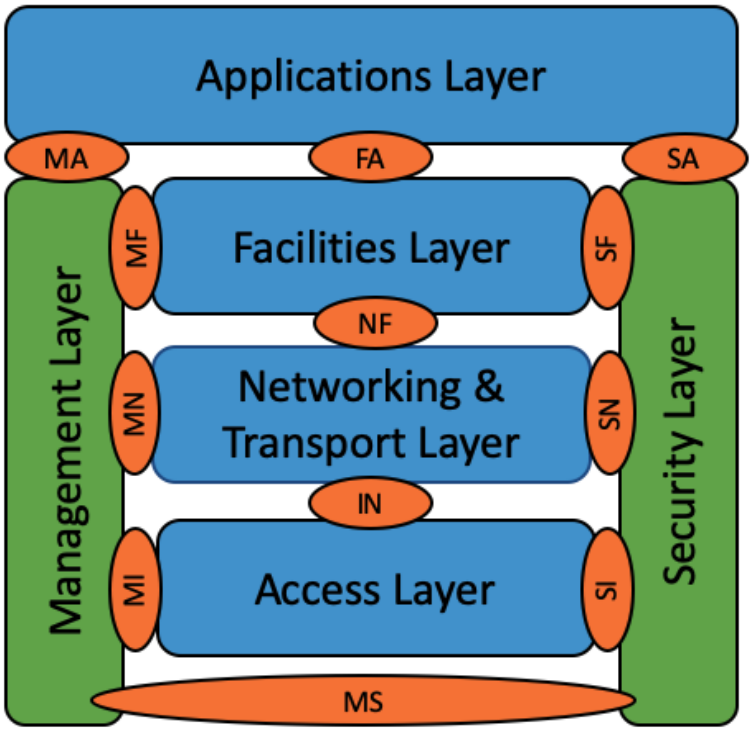


FIGURE 2.1: Architecture des ITS

Cette architecture est représentée à la Figure 2.1, elle comporte 4 couches horizontales : la couche d'accès, la couche réseau et transport, la couche service et la couche application. Où chacune de ces couches horizontales interagies avec la couche basse qui la précède et la couche haute qui la suit. L'architecture ITS comporte aussi 2 couches verticales qui interagissent avec toutes les couches horizontales : la couche sécurité et la couche management.

La couche management

Elle permet de gérer les fonctionnalités internes de la station ITS, notamment les fonctionnalités disponibles dans chaque couche, de déterminer quelles sont les technologies d'accès disponibles dans un lieu et dans un instant donné et aussi de gérer au mieux les flux de données en fonction des besoins de communication des applications.

La couche sécurité

Elle assure la confidentialité et la sécurité des communications au niveau de toutes les couches, en utilisant différents protocoles notamment : le chiffrement et l'authentification.

La couche d'accès

Cette couche intègre toutes les technologies d'accès existants et à venir, à condition que chaque technologie respecte les règles d'intégration à l'architecture spécifiée dans la norme (ISO 21218). Elle peut supporter le WiFi véhiculaire (IEEE 802.11p/bd), le WiFi urbain (IEEE 802.11n/b/g/ac), les réseaux cellulaires (2G, 3G, 4G et 5G), l'infrarouge, les ondes millimétriques, et les réseaux de capteurs 6LoWPAN (IEEE 802.15.4).

La couche réseaux et transport

Elle assure à la fois des communications localisées (ISO FNTP, ETSI GeoNetworking, IPv6) pour les échanges directs entre véhicules et entre véhicules et infrastructures (V2V, V2I) et aussi des communications distantes (IPv6) avec les pairs éloignés. FNTP et GeoNetworking sont conçus pour utiliser uniquement le WiFi véhiculaire, en revanche IPv6 permet la transmission quelles que soient les technologies localement disponibles.

La couche services

Elle permet aux applications de bénéficier des services mutualisés. En particulier une messagerie normalisée (transmission, réception et abonnement de messages contenant des informations variées), une base de données (LDM), des services de marquage des data-grammes (time-stamping, geo-stamping), de positionnement fiable (fusion de données émanant de plusieurs sources : GPS, balises routières, bus CAN, etc.). Les fondations de ces services mutualisées sont spécifiées dans la norme ISO 17429. Cette couche dispose aussi de la capacité d'orienter les data-grammes vers la pile de communication la plus adaptée en fonction du profil de communication et des capacités actuelles de la station ITS. Ainsi, un message de détection de danger pourra être transmis en utilisant soit la pile de protocoles non-IP pour des alertes diffusées aux véhicules proches, ou la pile IPv6 pour les alertes transmises à des serveurs, et cela sans que l'application n'effectue le moindre traitement.

La couche applications

Elle représente un siège de toutes les applications, qui peuvent bénéficier des services mutualisés ou de communication des couches inférieures. Pour cela, les applications doivent faire connaître leurs besoins de communication en fournissant à l'entité de gestion les caractéristiques de chacun des flux de données susceptibles d'être transmises par l'application. Les applications qui utilisent ces services sont moins complexes à développer, elles ne nécessitent pas pour son développeur des connaissances dans les protocoles et technologies de communication. Ces applications sont dédiées aux usagers car elles leur offrent la possibilité de portabilité.

2.3 Les systèmes de transports intelligents coopératifs (C-ITS)

Les systèmes ITS coopératifs se basent sur les informations issues de l'interaction directe entre véhicules et entre véhicules et infrastructures routières (V2I, V2V). Au niveau européen, de nouvelles structures ont été créées pour développer des normes (ETSI) spécifiques pour ces systèmes. Ils permettent aux usagers de la route et aux gestionnaires de trafic de partager des informations et de les utiliser pour coordonner leurs actions. Cet élément coopératif activé par la connectivité numérique entre les véhicules et entre les véhicules et l'infrastructure pour but d'améliorer considérablement la sécurité routière, l'efficacité du trafic et le confort de conduite, pour aider le conducteur à prendre les bonnes décisions et à s'adapter à la situation du trafic. Ces systèmes connectés permettront d'améliorer la qualité de l'air et diminuer le taux de pollution. Les sites C-ITS communiquent en utilisant des données des réseaux cellulaires et/ou le wifi IEEE 802.11p.

2.4 Les communications véhiculaires (V2X)

Les véhicules sont équipés de capteurs pour détecter des événements tels qu'une route glissante, un frein d'urgence, etc. De plus afin de pouvoir transmettre les informations aux autres véhicules via les communications V2V et à l'opérateur routier à travers les communications V2I, les véhicules doivent être équipés d'une OBU (On Board Unit). D'un autre côté l'opérateur routier peut également transmettre des informations (travaux routiers, etc.) aux véhicules (communications I2V) via les unités installées au bord de la route RSU (Road Side Unit).
D'autres types de communications sont envisagés pour être utilisés dans les zones urbaines, qui consistent à transmettre les informations des véhicules aux piétons (V2P).

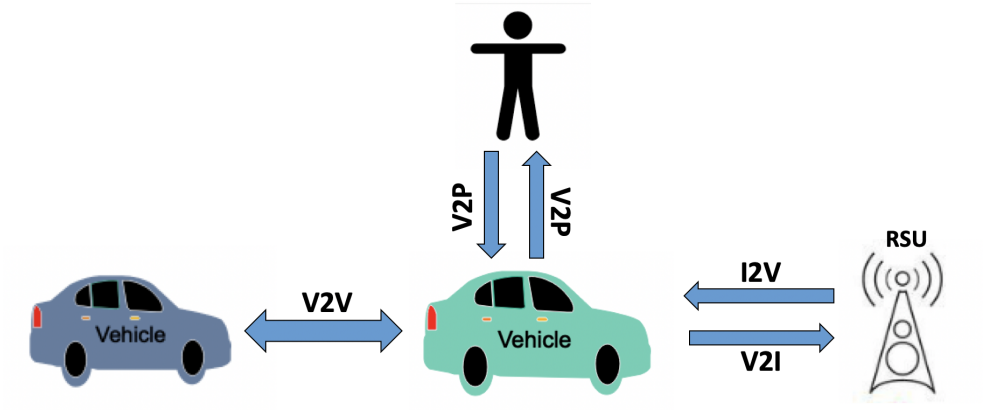


FIGURE 2.2: Différents types de communications V2X

2.5 Projets européens dédiées aux C-ITS

De nombreux projets majeurs ont été lancés par l'union européenne et en collaboration avec l'institut européen de standardisation et télécommunication ETSI (European Telecommunications Standards Institute). Ces projets ont été adoptés pour la mise en place des bases d'un système de transport intelligent coopératif(C-ITS). Parmi ces projets on peut citer : Scoop, C-Road, InterCor et Indid. Notre équipe est partenaire des deux derniers, mais elle participe aux autres projets via le consortium "C-ITS France". Les travaux de cette thèse s'inscrivent dans le cadre des projets InterCor Indid et la chaire RIVA.

2.5.1 Le projet Scoop

SCOOP est un projet pilote pour le déploiement de système de transports intelligents coopératifs, qui a été lancé en 2014. Il fédère de nombreux partenaires publics et privés autour du ministère l'écologie, du développement durable et de l'énergie, qui agit en tant que coordinateur : collectivités territoriales, opérateurs routiers, constructeurs automobiles PSA et Renault, universités et instituts de recherche. [49]
SCOOP a pour objectif de déployer 3000 véhicules sur 2000 km de routes, sur cinq sites : Ile-de-France, autoroute Paris-Strasbourg, Isère, le périphérique de Bordeaux, Bretagne. Ces sites se caractérisent par une grande diversité de types de routes (autoroutes, routes structurantes dans la zone métropolitaine, routes bidirectionnelles interurbaines et locales). Depuis janvier 2016, un opérateur de télécommunications , un prestataire de services de confiance et des partenaires autrichiens, espagnols et portugais ont rejoint le projet [49].

2.5.2 Le projet InterCor

InterCor est un projet européen sur trois ans (2017-2020) et avec un budget d'environ 30 millions d'euros, regroupant la France, les Pays-Bas, la Belgique et le Royaume-Uni. Il a pour but de connecter le transport routier en Europe (Figure 2.3). En effet, ce projet vise à harmoniser la mise en œuvre des C-ITS (les systèmes de transports intelligents coopératifs) et le déploiement stratégique des spécifications communes dans les quatre états membres. Les sites pilotes C-ITS, qui servent à communiquer des données des réseaux cellulaires et/ou ITS-G5, seront installés tout au long d'environ 1530 Km pour l'exploitation et l'évaluation des services C-ITS. Le projet interCor a pour objectifs de démontrer un déploiement interopérable à grande échelle des C-ITS à travers la France, les Pays-Bas, la Belgique et le Royaume-Uni pour parvenir à une mobilité plus sûre, plus efficace et plus économique des personnes et des biens.
Il vise également à fournir des services C-ITS à une échelle plus large en spécifiant, en utilisant et en favorisant une approche de communication hybride pour utiliser une combinaison de communication cellulaire et ITS-G5.

2.5.3 Le projet C-Road

C-Roads est une plate-forme qui regroupe les autorités routières et les opérateurs des États membres : France, Autriche, Belgique, République tchèque, Danemark, Finlande, Allemagne, Hongrie, Irlande, Italie, Portugal, Slovénie, Espagne, Suède, Pays-Bas, Royaume-Uni, Norvège, la Suisse et l'Australie. [13]
L'objectif de la plate-forme C-Roads est d'assurer la sécurité routière au niveau européen en alignant les spécifications des systèmes de transport intelligents coopératifs (C-ITS) pour garantir l'interopérabilité entre les ITS européen. Un déploiement rapide et à l'échelle de l'UE de services C-ITS harmonisés est la clé de cet objectif. Les États membres de C-Roads s'efforcent de garantir un fonctionnement sans failles des services C-ITS transfrontaliers et contribuent à ce titre à jeter les bases d'une conduite connectée et automatisée. [13]

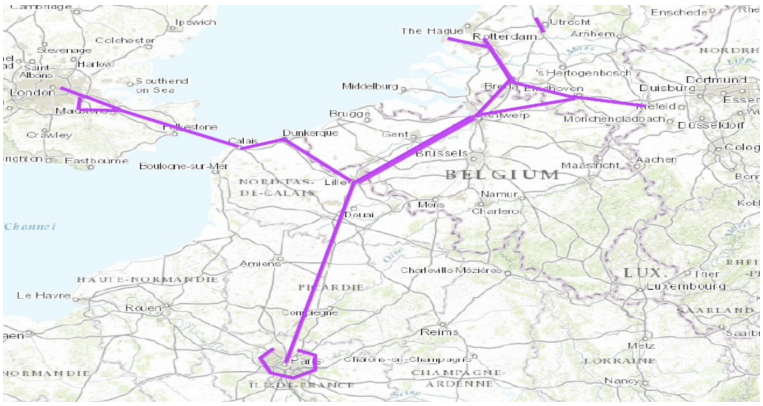


FIGURE 2.3: Déploiement du projet InterCor

Les déploiements actuels de C-ITS reposent sur les technologies de communication véhiculaires : ITS-G5 et les réseaux cellulaires 3G et 4G. En juin 2017, les membres de l'industrie automobile et de C-Roads ont déjà établi une coopération pour le déploiement des services C-ITS en Europe. Cette coopération vise à garantir l'interopérabilité des services en s'appuyant sur le protocole ITS-G5.

2.5.4 Le projet InDiD

InDiD est l'un des projets C-ITS porté par la France. Il a pour objectif de développer les systèmes de transport intelligents, il a été retenu par la Commission européenne dans le cadre d'appel à projets du CEF. Le projet est cofinancé à hauteur de 50% par l'Union Européen pour une durée de 5 ans (de 2019 jusqu'au 2023). Il s'inscrit dans la continuité des projets précédents C-ITS : SCOOP, C-ROADS et InterCor.
En plus de garantir une meilleure sécurité routière et une bonne gestion de trafic, le projet Indid consiste à développer de nouveaux cas d'usages destinés au milieu urbain, et aussi des cas d'usage de perception augmentée pour le véhicule autonome. En outre, il aborde la cartographie numérique haute définition de l'infrastructure. Il vise aussi les tests de communications véhiculaires basées sur la 5G pour le véhicule autonome.
Ce projet s'appuie sur un consortium fort, rassemblant 24 partenaires, répartis en France : des directions interdépartementales des routes, des acteurs industriels, des sociétés d'autoroutes et aussi des partenaires académiques (universités et centres de recherche).
InDiD vise à poursuivre le déploiement des C-ITS sur de nouveaux sites d'expérimentation routiers afin d'étendre la couverture des services offerts par l'infrastructure.
Les sites-pilotes se situent sur 4 grands bassins géographiques français, sur le versant méditerranéen, dans le sud-ouest, dans le centre et dans le nord de la France.

2.6 Technologies de communications véhiculaires

2.6.1 IEEE 802.11p (ITS-G5)

La norme IEEE 802.11p est une variante du standard IEEE 802.11 qui prend en charge les applications de Système de Transport Intelligent (ITS). Cela comprend l'échange de données entre les véhicules à grande vitesse et entre les véhicules et l'infrastructure routière (V2V, V2I), dans la bande ITS autorisée de 5,9 GHz (5,85-5,925 GHz). C'est aussi la base d'une norme pour les communications véhiculaires connue sous le nom ETSI ITS-G5 en Europe et WAVE aux USA.

2.6.1.1 Architecture ITS-G5

Elle est définie par la couche basse dans l'architecture en couches des ITS (Figure 2.1), qui est la couche d'accès. L'architecture de la couche d'accès ITS-G5 est représentée à la figure au-dessous :

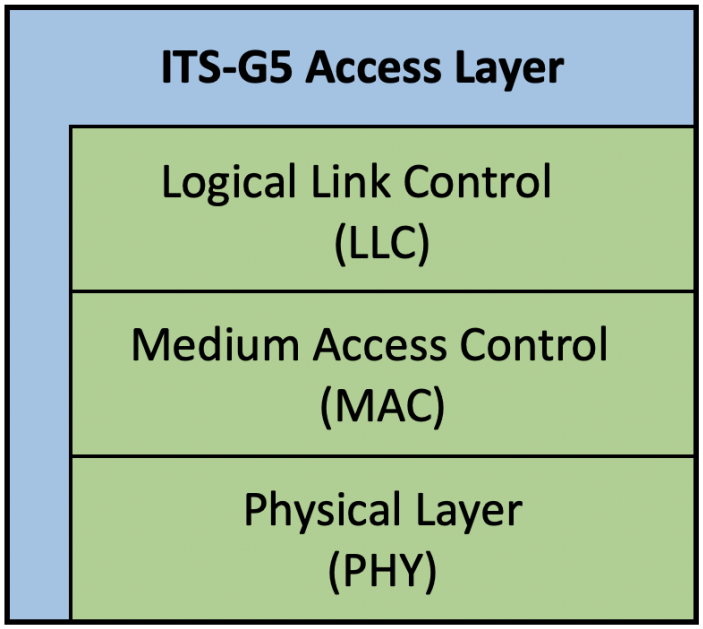


FIGURE 2.4: Couche d'accès d'ITS-G5

La couche d'accès ITS-G5 doit être conforme aux spécifications suivantes :

1. Utilisation de la forme d'onde OFDM à la couche physique.

2. La fonctionnalité de la couche MAC doit être telle qu'elle est définie dans le protocole IEEE 802.11 en définissant le paramètre MIB : dot11OCBActivé pour authentifier la communication en dehors du contexte d'un ensemble de services de base (BSS).

3. La couche contrôle de liaison logique (LLC) comme indiqué au standard ANSI/IEEE Stds 802.2c, où le mode de fonctionnement est introduit au Type 1 (mode sans connexion non reconnue).

La couche physique (PHY)

La couche physique représente une interaction entre la couche MAC et le support qui permet d'envoyer et de recevoir les trames. Elle est composée de deux sous-couches : la sous-couche PLCP (Physical Layer Convergence Protocol) et la sous-couche PMD (Physical Meduim Acees).

— **Sous-couche PLCP (Physical Layer Convergence Protocol) :** La sous-couche PLCP est une procédure de convergence qui permet de transformer la PDU (Packet Data Unit) arrivant de la couche MAC sous forme d'une trame OFDM. PLCP assure la communication avec la couche MAC.

— **Sous-couche PMD (Physical Medium Dependant) :** La sous-couche PMD représente une interface directe avec le médium, c'est-à-dire elle a un contact direct avec le support de transmission (le canal radio). Son rôle consiste à gérer l'encodage des données et effectuer les modulations.

En outre, la couche PHY dans 802.11p est l'OFDM. L'idée de base est de diviser le spectre de fréquences disponible en sous-canaux plus étroits (sous-porteuses). Le flux de données à débit élevé est divisé en un certain nombre de flux de données à débit inférieur transmis simultanément sur un certain nombre de sous-porteuses, chaque sous-porteuse étant étroitement liée. Il existe 52 sous-porteuses, dont 48 sont utilisées pour les données et 4 pilotes. La couche OFDM-PHY prend en charge huit taux de transfert différents, qui sont obtenus en utilisant différents schémas de modulation et taux de codages (Tableau 2.1).

Taux de transfert [Mbit / s]	Modulation	Taux de codage	Bit de données par symbole OFDM	Bits codés par symbole
3	BPSK	1/2	24	48
4.5	BPSK	3/4	36	48
6	QPSK	1/2	48	96
9	QPSK	3/4	72	96
12	16-QAM	1/2	96	192
18	16-QAM	3/4	144	192
24	64-QAM	2/3	192	288
27	64-QAM	3/4	216	288

Table 2.1: Modulation et taux de codage de la couche OFDM-PHY

2.6.1.2 Le paquet IEEE 802.11p

Le paquet résultant de la couche PHY est représenté dans la figure ci-dessous :

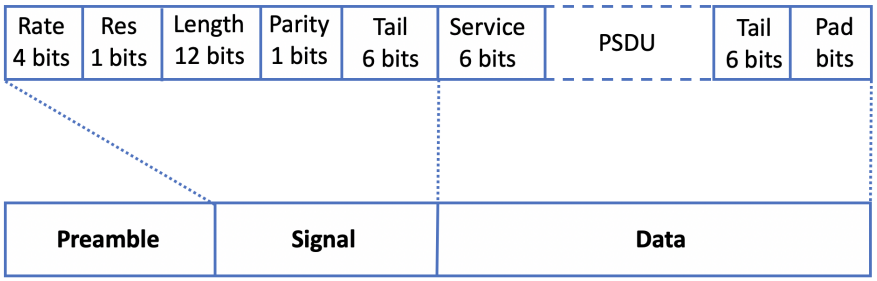


FIGURE 2.5: Structure de paquet de la couche PHY

Il est constitué de 3 champs : le champ préambule, le champ signal et le champ de données.

Champ préambule

Ce champ se compose de 10 séquences d'apprentissage courtes (STS) et de 2 séquences d'apprentissage longues (LTS), utilisées pour la détection de trame, l'alignement de trame et l'estimation de canal.

Champ signal

Le champ signal permet de déterminer le type de modulation à effectuer. Il représente un symbole OFDM qui sera transmis en utilisant la modulation BPSK avec un taux de codage R=1/2 et un débit de 3 Mbit/s. Par conséquent, la durée d'un symbole OFDM est de 8 microsecondes en 802.11p. La structure du champ signal est organisée comme suit : Rate, Reserved, Length, Parity et Tail (Figure 2.5).

— **Rate :** Le rôle du champ Rate est d'indiquer le débit de transmissions. Ce dernier est donné par les bits R1, R2, R3 et R4 comme suit :

Bits (R1—R4)	1101	1111	0101	0111	1001	1011	0001	0011
Débit (Mbps)	3	4.5	6	9	12	18	24	27

TABLE 2.2: Codage du débit

— **Reserved :** C'est un bit réservé pour une utilisation future.

— **Length :** Le champ longueur (Length) est un entier non signé formé de 12 bits indiquant le nombre d'octets dans le PSDU.

— **Parity :** Le bit P est un bit de parité positive sur les 16 premiers bits.

— **Tail :** Les 6 derniers bits sont mis à 0 et représentent la fin du champ signal.

Champ de données (Data)

La partie DATA est composée des sous-champs : Service, PSDU, Tail et les bits de bourrage (PAD bits).

— **Service :** Le sous-champ service de la norme IEEE 802.11p est représenté par une suite de 16 bits numérotés de 0 - 15. Les bits de 0 à 6 du champ service seront transmis en premier lieu. Ils sont fixés à zéro et utilisés pour la synchronisation au niveau du récepteur. Les neuf autres bits du champ service (7-15) doivent être réservés pour une utilisation future. Ces derniers bits sont mis à zéro.

— **PSDU :** Il représente le sous-champ de données à transmettre, il comporte une taille dynamique.

— **Tail :** Le sous-champ Tail de DATA est composé d'une suite de six «0», qui sont nécessaires pour réinitialiser les six registres du codeur convolutif à l'état « zéro».

— **PAD bits :** Consiste à ajouter un nombre suffisant de « 0 » pour ajuster la longueur des bits de données, pour qu'elle soit un multiple du nombre de bit codé par symbole OFDM.

La couche MAC

La couche MAC de la IEEE 802.11p est composée de deux fonctions de coordination d'accès au medium sans fil. La DCF (Distributed Coordination Function), fonction de coordination distribuée. Et de la PCF (Point Coordination Function), fonction de coordination centralisée. La DCF est une méthode d'accès asynchrone dont l'implémentation est obligatoire pour tous les équipements IEEE 802.11. La fonction PCF est un mode sans contention pour lequel l'accès au médium des différentes stations est coordonné par le point d'accès.
La couche MAC utilise un algorithme qui décide quand un nœud est autorisé à transmettre en fonction de l'état actuel de la chaîne de transmission, dans le but de minimiser les interférences dans le système pour augmenter la probabilité de réception des paquets. L'algorithme MAC déployé par 802.11p est l'accès amélioré à la coordination répartie (EDCA). Il est basé sur la fonction de coordination distribuée de base (DCF) avec l'ajout des attributs QoS. La fonction DCF est un moyen d'accès multiple qui utilise un algorithme d'élimination des collisions (CSMA/CA).
Dans l'algorithme CSMA/CA, un nœud commence à écouter le canal avant la transmission, si le canal est perçu comme inactif pour une période d'écoute prédéterminée, le nœud peut commencer à transmettre directement. En revanche, si le canal est occupé pendant la période d'écoute, le nœud effectuera une procédure de recul, c'est-à-dire qu'il doit différer son accès selon une période de temps aléatoire.

2.6.1.3 Fréquences occupées par ITS-G5

La bande de fréquences spécifiée pour la norme ITS-G5 est de 5.855 GHz jusqu'à 5.925 GHz, cette bande est répartie en 5 sous-bandes suivant l'utilisation comme suit :

— **ITS-G5A :** Applications liées à la sécurité routière des ITS.

— **ITS-G5B :** Autres applications ITS (non liées à la sécurité).

— **ITS-G5C :** RLAN (BRAN, WLAN).

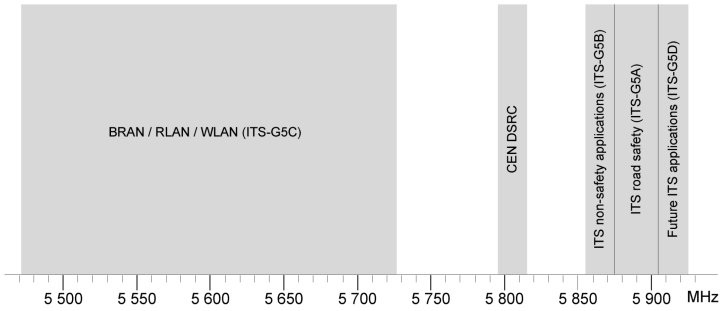


FIGURE 2.6: Attribution des canaux pour la norme ITS-G5 [22]

— **ITS-G5D :** Futures applications pour les ITS.

L'attribution des canaux doit être comme spécifiée dans le Tableau 2.3. On a un canal de contrôle (G5-CCH) et 7 canaux de services (de G5-SCH1 à G5-SCH7)

	Type de canal	**Bande de fréquence [MHz]**	**Numéro de canal**
ITS-G5A	G5-CCH	5 895 à 5 905	180
	G5-SCH2	5 885 à 5 895	178
	G5-SCH1	5875 à 5885	176
ITS-G5B	G5-SCH3	5865 à 5875	174
	G5-SCH4	5855 à 5865	172
ITS-G5C	G5-SCH7	5470 à 5725	94-145
ITS-G5D	G5-SCH5	5905 à 5915	182
	G5-SCH6	5915 à 5925	184

TABLE 2.3: Bandes de fréquences ITS-G5

2.6.2 IEEE 802.11bd

La norme IEEE 802.11bd représente la nouvelle génération des communications véhiculaires basées sur ITS-G5. Elle consiste à exploiter les techniques avancées des

couches PHY et MAC introduites dans 802.11n/ac/ax pour améliorer les performances de IEEE 802.11p [52]. Pour cela un groupe d'étude nommé IEEE 802.11 Next Generation V2X a été formé en mars 2018. Après une étude de faisabilité initiale, le groupe de travail IEEE 802.11bd a été créé en janvier 2019. Les objectifs de la création de la norme 802.11bd sont [70] :

— Atteindre le double du débit de 802.11p avec des vitesses relatives allant jusqu'à 500 km/h.

— Avoir une portée de communication deux fois plus grande que celle de la norme 802.11p.

— Avoir une forme de positionnement du véhicule en association avec les communications V2X.

En outre la norme 802.11bd doit supporter les fonctionnalités suivantes [52] :

— **Interopérabilité :** Les équipements 802.11p doivent être capables de décoder les transmissions des périphériques 802.11bd, et vice-versa.

— **Coexistence :** 802.11bd doit être capable de détecter les transmissions 802.11p et de différer l'accès aux canaux, et vice-versa.

— **Accès équitable :** dans les scénarios co-canal, 802.11bd et 802.11p doivent avoir l'égalité des chances d'accès aux canaux.

Les caractéristiques techniques de la norme 802.11bd sont résumées au tableau suivant :

Caractéristiques	**Spécifications Techniques**
Bandes de Fréquences	5.9 Ghz et 60 GHz
Codage Canal	LDPC
Re-transmission	Dépendant de la congestion
Élimination de l'effet Doppler	Ajout de Midambles
Espacement des sous-porteuses	312.5 Khz, 156.25 KHz, 78.125 KHz
Vitesse relative supportée	500 Km/h
Flux spatial	Multiple

Table 2.4: Caractéristiques techniques de la norme 802.11bd.

2.6.3 Les réseaux cellulaires

Les réseaux cellulaires reposent sur la technique des cellules qui sont des zones élémentaires de couverture qui s'interpénètrent et permettent de couvrir une zone à desservir d'un territoire.
Dans les réseaux cellulaires, chaque antenne couvre un territoire défini et lors des

déplacements de l'utilisateur, il change de cellule. Nous utilisons actuellement des technologies cellulaires du type 2G, 3G et 4G. Mais elles ne seront pas suffisantes pour plusieurs services hauts débits comme la vidéo haute définition et la réalité virtuelle, ce qui a nécessité l'émergence de la 5G, qui sera opérationnelle prochainement.
Actuellement, les centres de recherche et les opérateurs industriels ont lancé plusieurs projets de recherche afin de développer la nouvelle génération des réseaux cellulaires : 6G.

2.6.3.1 Le réseau cellulaire 4G/LTE

La 4G/LTE (Long Term Evolution) ou E-UTRAN (Evolved Universal Terrestrial Access Network), à été introduite par le consortium 3GPP dans la Release 8. Cette quatrième génération (4G) des réseaux cellulaires représente la partie accès de l'Evolved Packet System (EPS). Les principales exigences pour de la 4G sont : une efficacité spectrale et des débits de données élevés, un court temps de latence ainsi qu'une flexibilité de la fréquence et de la bande passante [53].
Le réseau cœur de LTE est, le EPC (Evolved Packet Core). Comme on le voit à la Figure 2.7, il est constitué [51] :

— D'une base de données centrale HSS (Home Subscriber Server).

— D'un ou plusieurs équipements gérant la localisation appelée MME (Mobility Management Entity).

— D'une ou plusieurs passerelle d'accès vers les réseaux tiers appelés PDN-GW, ou plus simplement PGW (Packet Data Network Gateway).

— De passerelles en plus grand nombre appelé SGW (Serving Gateway), chacune gérant une zone donnée (par exemple, une région).

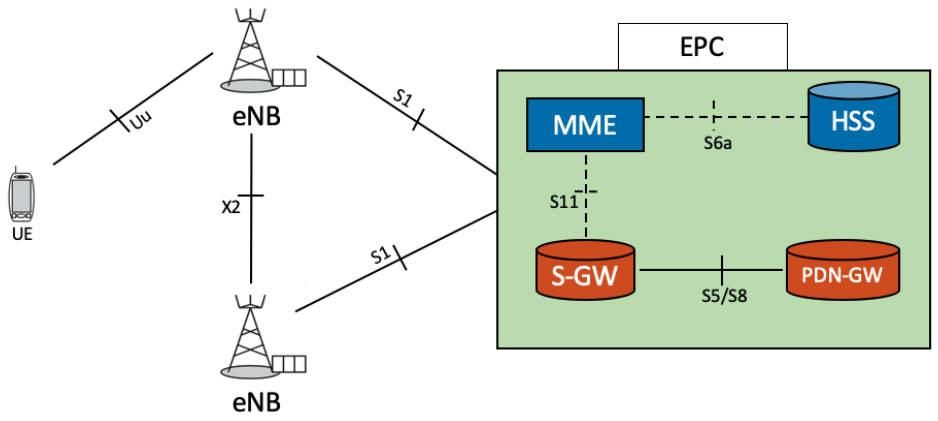


FIGURE 2.7: Architecture LTE.

Le réseau d'accès LTE est constitué de stations de base appelées eNodeB ou eNB (Evolved Node B) et le terminal appelé UE (User Equipment). Les eNB sont interconnectés entre eux via l'interface X2 et connectés avec le réseau central EPC par l'interface S1. Le terminal UE est connecté au réseau via l'interface Uu (Figure 2.7). Pour atteindre une efficacité spectrale élevée et permettre une planification efficace dans les domaines temporel et fréquentiel, une approche multi-porteuse pour un accès multiple a été choisie par le 3GPP. Donc la couche physique de la technologie LTE est basée sur l'OFDM, avec l'utilisation des modulations d'ordre supérieur (jusqu'à

64-QAM). Les spécifications techniques de l'OFDM utilisée pour LTE sont décrites au Tableau 2.5
LTE peut prendre en charge à la fois la technologie duplex par répartition dans le temps (TDD) et le duplex par répartition en fréquence (FDD). Les bandes de fréquences définies pour LTE allant de 700 MHz à 2,7 GHz, avec des largeurs de bande flexibles de 1,4 MHz jusqu'à 20 MHz.
Le débit de données de crête théorique le plus élevé sur le canal de transport est de 75 Mbps dans la liaison montante, et de 300 Mbps dans la liaison descendante.

Bande spectrale occupée (MHz)	1.4	3	5	10	15	20
Largeur d'une sous-porteuse (Δf) (kHz)	15					
Nombre de sous-porteuses	72	180	300	600	900	1200
Taille de la FFT	128	256	516	1024	1536	2048
Durée de symbole (μs)	66.67					

TABLE 2.5: Paramètres de l'OFDM pour LTE

2.6.3.2 La technologie 5G

La 5G est le réseau mobile de 5e génération, elle permet un nouveau type de réseau conçu pour connecter pratiquement tout le monde, y compris les machines, les objets et les appareils.
La 5G offre une amélioration importante par rapport aux capacités de la 4G, elle était définie initialement dans la Release 15 de 5G New Radio (NR). Les spécifications de cette Release se concentrent sur : Enhanced Mobile Broadband (eMBB) et Ultra-Reliable Low-Latency Communications (URLLC), pour atteindre des débits de données très rapides et fournir une très faible latence dans les communications sans fil.
La 5G NR devrait fonctionner avec la 4G et même utiliser le réseau central 4G pour les données et les plans de contrôle en mode non-standalone (NSA). Il est également prévu que la 5G, la 4G et le Wi-Fi coexisteront sur les mêmes opérateurs et utiliseront des bandes non-licenciées pour augmenter la capacité en dessous de 6 GHz.
La Release 15 de la 5G NR décrit la couche physique, qui est très importante car elle définit la structure qui compose le signal radio et la manière dont le signal est communiqué via l'interface radio.

Forme d'onde et fréquences allouées pour 5G NR

5G NR a défini l'OFDM comme étant la forme d'onde à utiliser dans la liaison descendante (DL) et la liaison montante (UL).
Contrairement à la 4G, l'OFDM spécifiée à la 5G-NR permet des espacements des sous-porteuses évolutifes. Ou, des espacements de sous-porteuses de 15, 30 et 60 kHz sont utilisés pour les bandes de fréquences inférieures, et des espacements de sous-porteuses de 60, 120 et 240 kHz sont utilisés pour les bandes de fréquences supérieures.
Un espacement plus grand des sous-porteuses aux fréquences plus élevées contribue également à la robustesse de la forme d'onde, car le bruit de phase intégré peut être un problème dans les conceptions mmWave.

En LTE, on a une bande passante qui peut atteindre un maximum de 20 MHz. En revanche, la 5G NR a une bande passante maximale jusqu'à 100 MHz en FR1 (jusqu'à 7,125 GHz), ou jusqu'à 400 MHz en FR2 (jusqu'à 52,6 GHz).
Les spécifications techniques de la forme d'onde OFDM 5G-NR sont décrites au tableau ci-après :

Fréquences	FR1 : 410 MHz – 7.125 GHz FR2 : 24.25–52.6 GHz
Bande passante maximale	FR1 : supérieure à 100 GHz FR 2 : supérieure à 400 MHz
Espacement des sous-porteuses	6 GHz : 15 kHz, 30 kHz, 60 kHz >6 GHz : 60 kHz, 120 kHz, 240 kHz
Nombre maximum de sous-porteuses	3300 (supérieure à 4096 FFTs)
Modulations	16QAM, 64QAM et 256QAM

Table 2.6: Spécifications de fréquences et de formes d'onde 5G NR Release 15

L'objectif de la 5G est d'atteindre un débit de 20 Gbit/s dans la liaison descendante (DL) et 10 Gbit/s dans la liaison montante (UL), pour prendre en charge de nouvelles applications telles que la diffusion en continu à haut débit de films 4K ou 8K UHD.
L'harmonisation du spectre entre les régions est nécessaire pour offrir la gamme complète de capacités et de couverture aux utilisateurs du monde entier. La 5G NR spécifie une fréquence jusqu'à 52,6 GHz et de nouvelles bandes d'exploitation qui ouvrent près de 10 GHz de nouveau spectre :

— **FR1 :** 410 MHz à 7,125 GHz, avec l'ajout de 1,5 GHz de nouveau spectre dans les bandes de fréquences : 3,3–4,2 GHz, 3,3–3,8 GHz, 4,4–5 GHz.

— **FR2 :** 24,25 à 52,6 GHz, avec l'ajout de 8,25 GHz de nouveau spectre dans les bandes de fréquences : 26,5–29,5 GHz, 24,25–27,5 GHz, 37–40 GHz.

Des études et des essais ont eu lieu dans plusieurs régions sur certaines bandes de fonctionnement dans des fréquences inférieures et supérieures à 6 GHz. Ils commencent à faire des lancements initiaux, comme suit :

— En dessous de 1 GHz, il existe plusieurs bandes d'intérêt dans 600, 700 et 800 MHz qui peuvent prendre en charge les services IoT.

— 1–6 GHz sera utilisé pour augmenter la couverture et la capacité.

— Au-dessus de 6 GHz sera principalement utilisé pour répondre au besoin de cas d'utilisation à très haut débit.

— la 5G NR a spécifié une gamme de fréquences allant jusqu'à 52,6 GHz, tandis que des études sont en cours pour que les futures versions fonctionnent dans

la gamme de fréquences 64–76 GHz.

— Un objectif principal en Chine, en Europe, en Corée et au Japon est la plage 3,3–3,8 GHz qui peut être utilisée pour différents services de la 5G . 4,4 à 4,9 GHz sont également envisagés en Chine et au Japon.

— Les cibles initiales mmWave sont 28 GHz et 39 GHz au Japon et aux États-Unis.

Architecture de la 5G

La principale caractéristique de la 5G est l'introduction d'une nouvelle interface radio, la New Radio (NR), qui offre la flexibilité nécessaire pour prendre en charge les nouveaux services. Une autre caractéristique clé de la 5G est que le réseau d'accès 5G peut se connecter non seulement à un nouveau réseau central 5G, mais également au réseau central 4G (LTE). De ce fait deux options de déploiement sont définies pour l'architecture de la 5G :

1. **L'architecture ≪Non-Stand Alone≫ (NSA) :** où le réseau d'accès radio 5G (AN) et sa nouvelle interface radio (NR) sont utilisés en conjonction avec l'infrastructure existante LTE et EPC Core Network, ce qui rend la technologie NR disponible sans remplacement du réseau. Dans cette configuration, seuls les services 4G sont pris en charge, mais profitant des capacités offertes par la Nouvelle Radio 5G (une latence plus faible). La NSA est également connue sous le nom de ≪E-UTRA-NR Dual Connectivity (EN-DC)≫ ou ≪Architecture Option 3≫.
   L'architecture NSA est représentée à la figure ci-après :
   L'architecture NSA peut être considérée comme une étape temporaire vers

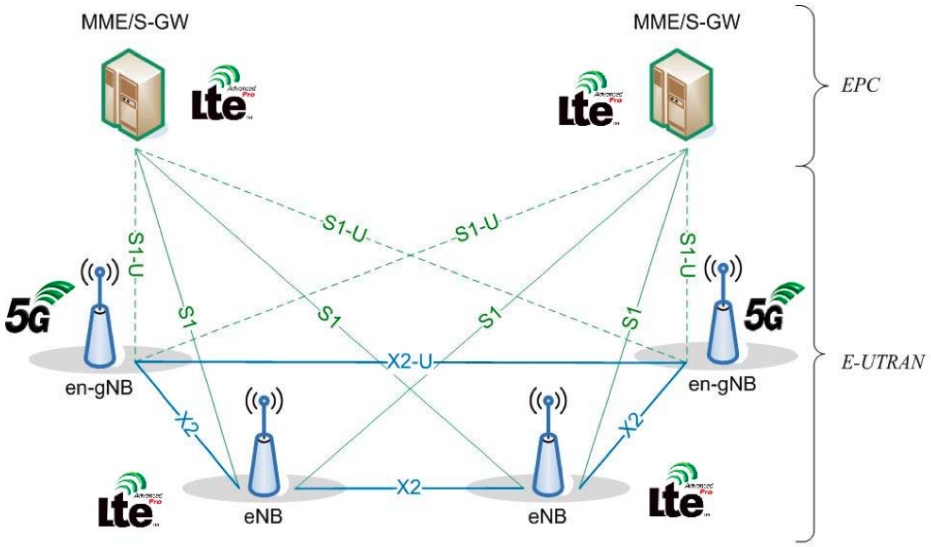


FIGURE 2.8: Architecture NSA 5G

le déploiement complet de la 5G. Dans l'architecture NSA, la station de base (5G) NR (nœud gNodeB "gNB") se connecte à la station de base (4G) LTE (nœud "eNB") via l'interface X2. L'interface X2 a été introduite avant pour connecter deux eNB. La Release 15 prend également en charge la connexion d'un eNB et d'un en-gNB pour fournir NSA.

2. **L'architecture ≪Stand-Alone≫ (SA) :** où le NR est connecté au 5G-CN (Core Network). Dans cette configuration, l'ensemble complet des services 5G

est pris en charge. L'architecture SA est représentée à la Figure 2.9 L'archi-

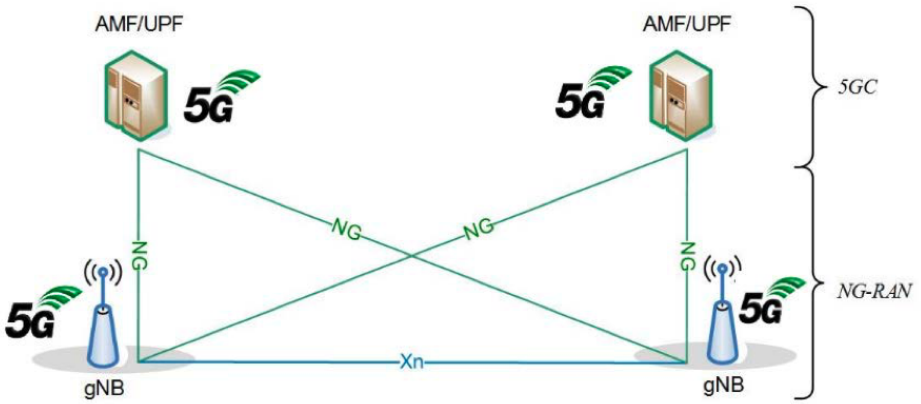


FIGURE 2.9: Architecture SA 5G

tecture SA peut être considérée comme le déploiement complet de la 5G, où, aucune partie du réseau 4G n'est nécessaire pour fonctionner. Les stations de base NR (gNB) se connectent entre elles via l'interface Xn, pour former un réseau d'accès appelé "NG-RAN". Ce réseau se connecte au réseau 5GC ou 5GCN en utilisant l'interface NG.

L'architecture du 5G AN est extrêmement simple puisqu'elle consiste en une seule entité, qui est le gNB, qui se connecte au 5G CN via l'interface NG. Il peut également se connecter à un autre gNB via l'interface Xn et à l'eNB de la 4G via l'interface X2. L'architecture 5GC repose sur un cadre dit «d'architecture basée sur les services» (SBA), où les éléments d'architecture sont définis en termes de «fonctions de réseau» (NF). L'architecture de base du système 5G (SA, sans Roaming) est illustrée ci-dessous :

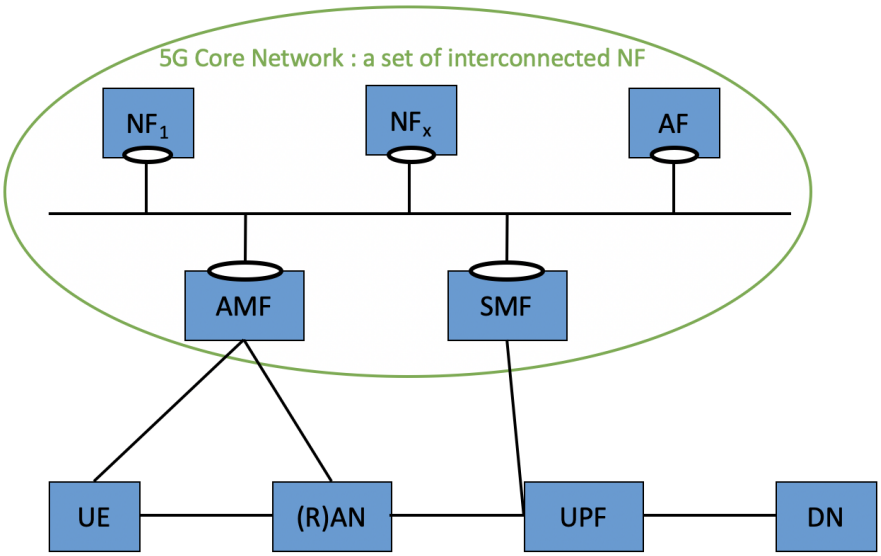


FIGURE 2.10: Architecture générale 5G

Dans la Figure 2.10 seuls les éléments et les fonctions réseaux essentiels sont indiqués :

- **UE (User Equipment) :** Utilisateur terminal
- **(R)AN (Radio) Access Network :**Réseau d'accès G5 AN, qui est constitué des stations de bases gNB interconnecté entre eux.
- **UPF (User Plane Function) :** responsable de : Point de session externe PDU d'interconnexion au réseau de données, routage et transfert de paquets, Inspection des paquets et partie plan utilisateur de l'application des règles de politique, classificateur de liaison montante pour prendre en charge le routage des flux de trafic vers un réseau de données, point de branchement pour prendre en charge la session PDU multi-résident, gestion QoS pour le plan utilisateur.
- **DN (Data Network) :** il s'agit de :services d'opérateur, accès Internet ou services tiers.
- Les fonctions réseau (NFs) essentielles du réseau 5GC représentées à la Figure 2.10 sont :
  - **AMF (Access and Mobility Management Function) :** Elle est responsable de : la gestion des inscriptions, la gestion des connexions, la gestion de l'accessibilité, la gestion de la mobilité, l'interception légale, l'authentification et l'autorisation d'accès, la gestion des services de localisation et la notification des événements de mobilité UE.
  - **SMF (Session Management Function) :** responsable de : la gestion de session, par ex. Établissement, modification et libération de session, allocation et gestion d'adresses IP UE, fonctions DHCPv4 et DHCPv6 (serveur et client), proxy ARP, sélection et contrôle de la fonction UP, y compris le contrôle de l'UPF, le routage du trafic UPF et la fonctionnalité Roaming
  - **AF (Application Function) :** Cette fonction interagit avec le réseau principal 3GPP afin de fournir des services tels que l'influence des applications sur le routage du trafic, l'accès à la fonction d'exposition du réseau et l'interaction avec le cadre de politique pour le contrôle des politiques

2.6.4 Réseau cellulaire V2X (C-V2X)

C-V2X ou LTE-V2X est une technologie d'accès radio pour les communications V2X, développée par 3GPP dans sa Release 14. Les ressources temps-fréquence utilisées par C-V2X sont similaires à celles de LTE.
Dans la technologie C-V2X, il existe deux modes de fonctionnement pour les communications V2X, comme illustré à la Figure 2.11 :

- **Communication V2X via l'interface LTE-Uu :** LTE-Uu est l'interface traditionnelle entre un eNB et un équipement utilisateur (UE). Tout UE utilisant l'interface LTE-Uu doit transmettre son message à l'eNB dans la liaison montante, ainsi l'eNB transmet son message à l'UE de destination dans la liaison descendante. En utilisant l'interface LTE-Uu, un UE peut transmettre un paquet à l'eNB dans la liaison montante. Le même eNB ou un autre eNB peut transmettre ce paquet à un UE éloigné en utilisant une liaison descendante.

- **Communication V2X via l'interface PC5 :** L'interface aérienne PC5 permet des communications directes entre les UEs sans que chaque paquet passe par l'eNB, en utilisant la liaison *sidelink*. L'interface PC5 comprend deux modes : sidelink mode 3 et sidelink mode 4.

  1. **C-V2X sidelink mode 3 :** Ce mode est défini pour les scénarios où la couverture de l'eNB est disponible. Dans ce mode l'allocation des ressources pour la transmission est gérée par l'eNB.

  2. **C-V2X sidelink mode 4 :** Ce mode est défini pour être utilisé lorsque les

UE sont en dehors de la couverture cellulaire (pas de eNB disponible). Dans ce mode les UEs réservent les ressources de manière autonome à l'aide de l'algorithme *resource reservation*. Cet algorithme nécessite que chaque UE détecte le canal pendant 1 seconde et traite les résultats de détection afin de garantir que les UEs voisins sélectionnent et réservent des ressources orthogonales, afin de minimiser les collisions de paquets.

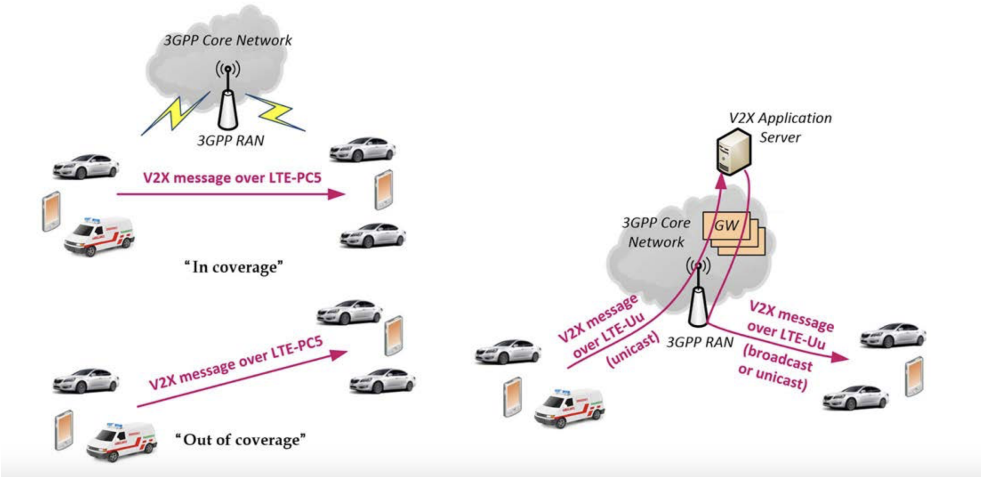


FIGURE 2.11: Communication V2X via l'interface PC5 et l'interface LTE-Uu

2.6.4.1 Architecture C-V2X

L'architecture du réseau C-V2X est représentée à la Figure 2.12. La «Fonction de contrôle V2X», est la fonction logique définie pour les actions liées au réseau requis pour les communications V2X et effectue l'autorisation et la fourniture des paramètres nécessaires pour la communication V2X des UEs.

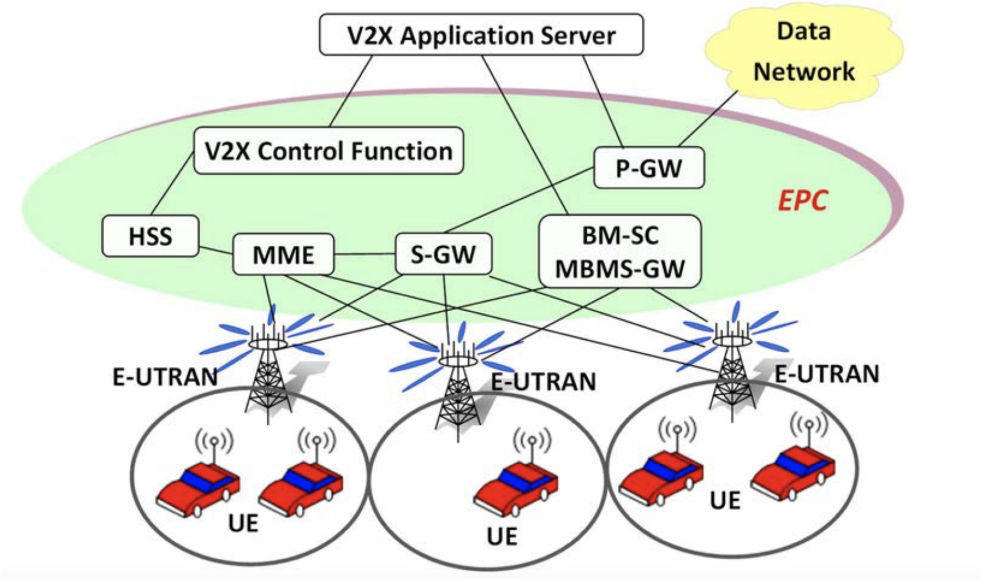


FIGURE 2.12: Architecture globale C-V2X

D'après l'architecture C-V2X illustrée à la Figure 2.12, on constate que cette architecture comporte des blocs avec des fonctionnalités dédiées aux applications V2X.

— **V2X Control Function :** Cette fonction de contrôle V2X est la fonction logique utilisée pour les actions liées au réseau requis pour les communications V2X. Elle est utilisée pour fournir à l'UE les paramètres nécessaires liés aux services V2X, lorsque l'UE est en couverture ou hors couverture de l'E-UTRAN.

— **V2X Application Server :** Le serveur d'applications V2X (V2X AS) peut prendre en charge les fonctionnalités suivantes :

  — Réception et envoie des données dUE dans une zone cible via le mode unicast ou à travers le MBMS
  — Mappage des informations de localisation géographique.
  — Fournir les USD V2X pour que l'UE reçoive le trafic V2X basé sur MBMS vers la fonction de contrôle V2X.
  — Fournir les paramètres pour les communications V2X sur le point de référence PC5 à la fonction de contrôle V2X.
  — Envoyer les informations MBMS aux BM-SC.
  — Demander aux BM-SC d'activer, de désactiver ou de modifier le support MBMS.

— **MME :** En plus des fonctions définies précédemment, dans le cas de V2X, le MME exécute les fonctions suivantes :

  — Obtenir les informations d'abonnement liées à V2X dans le cadre des données d'abonnement.
  — Fournit une indication à l'E-UTRAN sur l'état d'autorisation UE lors de l'utilisation de V2X.

— **BM-SC (Broadcast Multicast Service Centre) :** Dans le cas de V2X, le BM-SC remplit les fonctions suivantes :

  — Recevoir les informations MBMS du serveur d'applications V2X.
  — Envoyer les informations MBMS aux MBMS-GW.

— **MBMS (Multimedia Broadcast/Multicast Service) :** Dans le cas de V2X, le MBMS, s'en charge de l'allocation des adresses de multi-diffusion IP.

2.6.5 NEW RADIO NR-V2X

NR-V2X est la nouvelle version des réseaux cellulaires définie par 3GPP dans la Release 16, elle consiste à développer des solutions techniques, qui prennent en charge les cas d'utilisation de la 5G-V2X [52].
NR-V2X est conçu pour compléter et inter-fonctionner avec LTE-V2X en prenant en charge des cas d'utilisation plus avancés. Il existe 25 cas d'utilisation avancés de V2X, qui peuvent être regroupés en quatre domaines :

— Les capteurs étendus qui permettent l'échange de données de capteur et de vidéo en direct entre les véhicules, les piétons, les unités d'infrastructure et les serveurs d'applications V2X pour étendre la perception de l'environnement à l'UE.

— La conduite avancée permet une conduite autonome ou semi-autonome en échangeant des données de capteur et une intention de conduite qui permet aux véhicules de coordonner leurs trajectoires.

— La conduite à distance permet à un conducteur à distance ou à une application V2X de conduire à distance un véhicule pour des passagers qui ne peuvent pas conduire eux-mêmes ou des véhicules conduits dans des environnements dangereux. La principale exigence pour la conduite à distance est la communication à faible latence.

— Platooning de véhicules, qui représente l'aptitude d'un groupe de véhicules voyageant ensemble à s'organiser. Où un véhicule pilote en tête fournit des messages aux autres véhicules tout en permettant des distances entres véhicules plus petites.

La couche PHY de NR-V2X est similaire à celle du réseau 5G, définie dans la Release 15 [52]. Elle repose sur la forme d'onde OFDM, avec un espacement des sous-porteuses flexible : 15, 30 et 60 kHz. Elle opère sur les fréquences inférieures à 6 GHz (FR1), et aussi sur les bandes de fréquences supérieures à 6 GHz (FR2).
NR-V2X comporte deux *sidelink* modes [52].

1. **NR-V2X sidelink mode 1 :** Il définit des mécanismes qui permettent des communications véhiculaires directes dans des zones de couvert par gNB. Dans ce mode, le gNB alloue des ressources aux UE.

2. **NR-V2X sidelink mode 2 :** Il prend en charge les communications véhiculaires directes dans le scénario hors couverture de gNB. Dans ce mode Chaque UE sélectionne sa ressource de manière autonome ou certains UE aident d'autres UE à effectuer la sélection des ressources.

2.6.5.1 Architecture NR-V2X

3GPP Rel.16 a proposé de nombreuses solutions alternatives pour l'architecture de référence de 5G-V2X [52].
La première architecture (Figure 2.13), repose sur un modèle de référence d'architecture 5G basée sur PC5 et Uu, en intégrant les fonctionnalités V2X. Dans cette alternative, comme la fonction de contrôle V2X ne gère que la configuration UE, il est proposé que la nouvelle procédure dans 5GC gère la configuration UE. Par conséquent, il est proposé d'utiliser PCF comme entité fonctionnelle de provisionnement de la politique/des paramètres V2X.

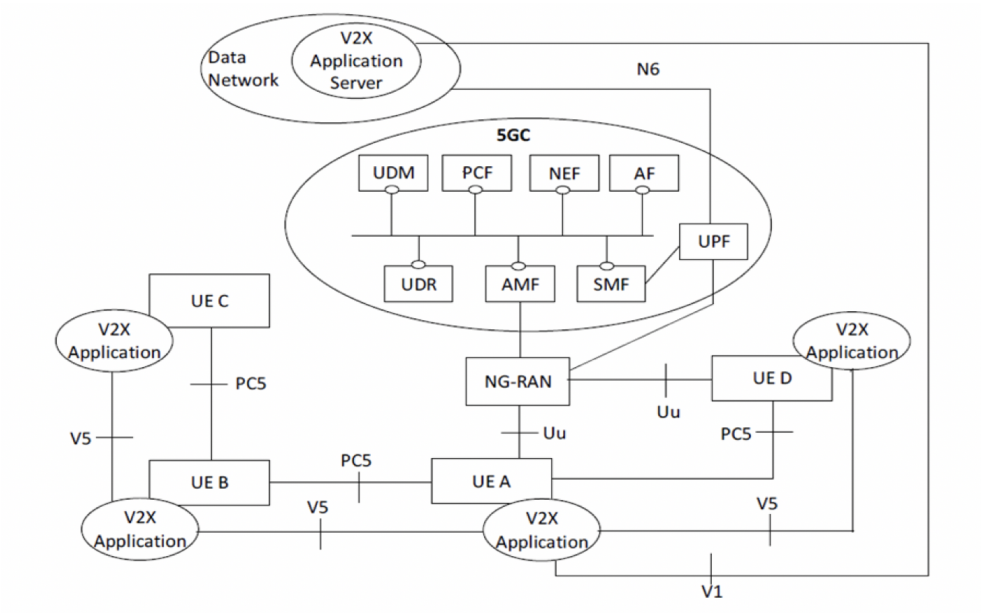


FIGURE 2.13: Architecture NR-V2X -Alternative 1-

La deuxième alternative(Figure 2.14), EST basée sur L'utilisation des interfaces PC5 et Uu. Dans cette proposition, V2X Control Function, V2X Application Server est le même que celui définit dans C-V2X. En outre, un serveur d'applications V2X peut se connecter à plusieurs PLMN.

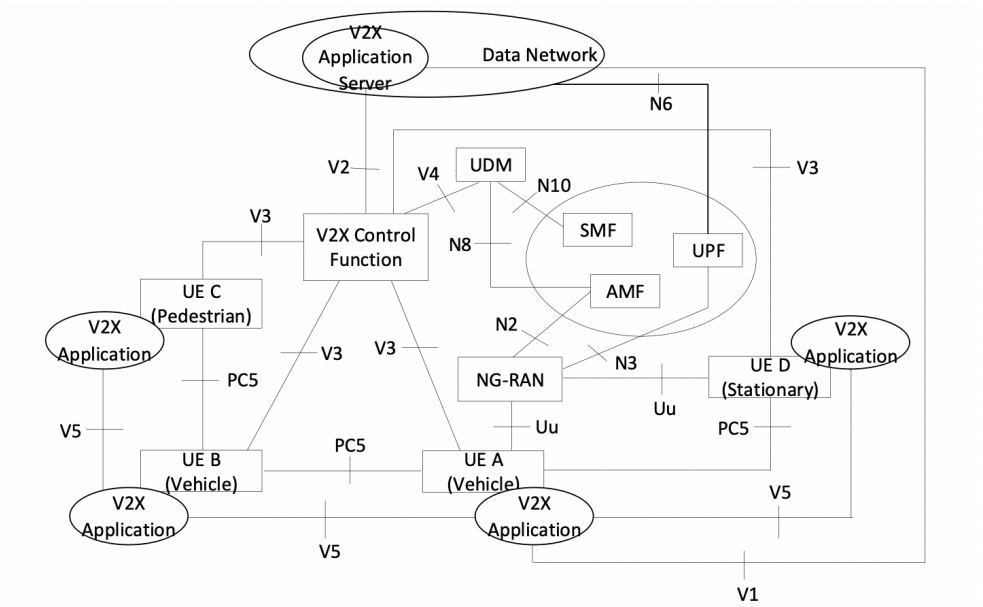


FIGURE 2.14: Architecture NR-V2X -Alternative 2-

La troisième alternative propose d'introduire la fonction de contrôle V2X en tant que nouvelle entité CP dans le 5GC comme indiqué à la Figure 2.15 ci-dessous. La fonction de contrôle V2X est utilisée pour fournir à l'UE les paramètres nécessaires afin d'établir les services V2X.

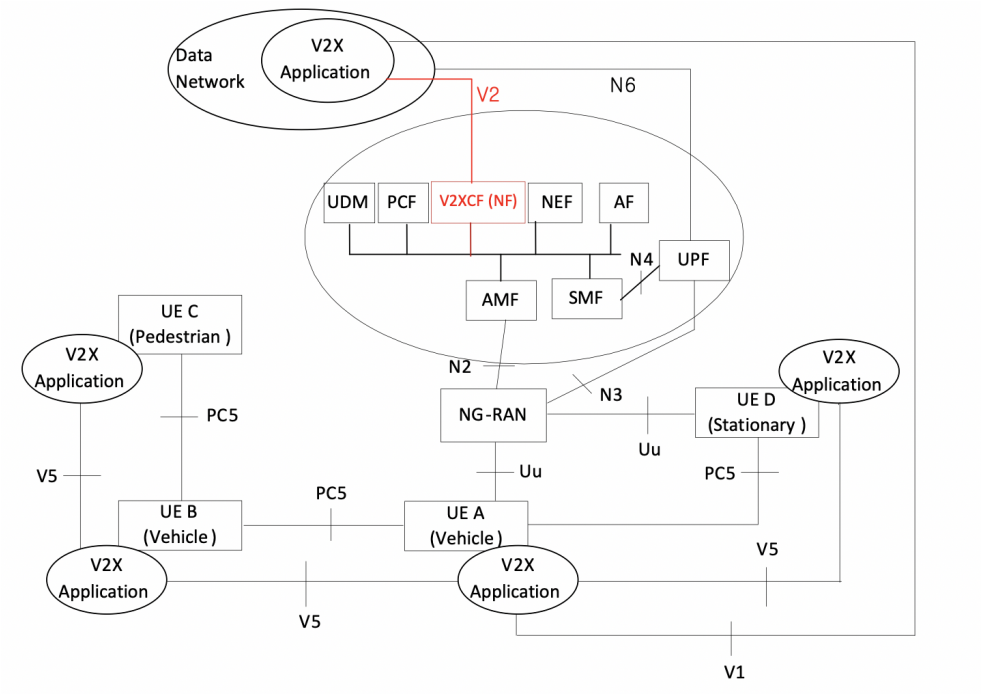


FIGURE 2.15: Architecture NR-V2X -Alternative 3-

2.7 Conclusion

Dans ce premier chapitre, nous avons d'abord présenté les systèmes de transport intelligent, ainsi que leurs architectures en couches. Après nous avons décrit en détail les fonctionnalités de couche la plus basse (couche physique) étant donné que notre travail se concentre essentiellement sur les techniques d'accès et les technologies de communications véhiculaires. Dans ce contexte, nous avons donné un aperçu de l'architecture et des caractéristiques techniques du wifi véhiculaire avec ces deux versions : IEEE802.11P et IEEE802.11bd.
Ensuite, nous avons introduit les réseaux cellulaires (4G,5G), et aussi nous avons décrit les nouvelles spécifications techniques ajoutées à ces réseaux cellulaires dans les dernières Releases, pour qu'ils soient en mesure de supporter les applications V2X, où ils ont spécifié les standards : C-V2X et NR-V2V comme réseaux dédiés aux communications V2X.

Chapitre 3

Estimation du canal véhiculaire

Sommaire

3.1 Introduction **40**
3.2 Modélisation d'un système de communication véhiculaire **41**
3.3 Canal de propagation véhiculaire **43**
3.3.1 Effet du canal sur les communications véhiculaires 45
3.4 Techniques d'estimation de canal **45**
3.4.1 Estimation de canal basée sur la CFR 45
3.4.2 Estimation de canal basée sur un modèle paramétrique . 46
3.4.3 Estimation de canal itérative (ICE) 48
3.5 Algorithmes proposés **50**
3.5.1 Algorithme SCR 50
3.5.2 Algorithme MSCR 51
3.6 Évaluation des performances **52**
3.6.1 Évaluation du SER vs. SNR 52
3.6.2 Temps de latence (temps d'exécution) 55
3.6.3 Intervalle de confiance (CI) 56
3.7 Conclusion **57**

3.1 Introduction

Étant donné que les véhicules connectés sont très mobiles lors de la communication, les signaux reçus sont fortement influencés par l'effet du canal de communication sans fil. Pour cette raison, l'estimation de canal est une tâche clé dans la communication véhiculaire. Par conséquent, de nombreux travaux de recherche ont été réalisés dans ce domaine.
Dans [88], une estimation itérative de canal variant dans le temps pour la technologie de communication véhiculaire (802.11p) est présentée. L'approche consiste à concevoir un modèle de canal de sous-espace basé sur des paramètres statistiques de second ordre connus. Ce modèle conçu vise à minimiser l'erreur quadratique moyenne MSE. Ainsi, le récepteur itératif pourra converger vers la réponse du canal parfaite. Les résultats démontrent que cette approche donne des performances prometteuses.
Dans [93], une méthode d'estimation de canal basée sur les décalages Doppler est introduite. Elle consiste à utiliser le taux de passage à niveau (LCR) avec un algorithme itératif spécifique afin de converger vers la valeur de décalage Doppler, ce qui se traduit par une faible MSE.
Une approche appelée estimation de canal par décision-dirigée (DD) utilisée dans l'environnement véhiculaire est présentée dans [54]. Cette approche se base sur une méthode d'estimation de canal par modèle paramétrique. Elle repose sur l'utilisation d'un filtre de Wiener 1D qui exploite la corrélation du canal dans le domaine fréquentiel. Cela peut offrir de bonnes capacités de suivi pour les canaux véhiculaires.
Dans [24], une nouvelle approche d'estimation de canal pour les réseaux ad hoc véhiculaires est décrite. Cette méthode étudie l'expression dérivée du taux d'erreur sur les bits (BER) en présence d'une estimation de canal imparfaite. En utilisant une allocation optimale aux pilotes et aux données, l'erreur d'estimation de canal peut être réduite.
Cependant, la plupart des méthodes explorées précédemment n'ont pas été évaluées sur un canal véhiculaire à évanouissement doublement sélectif, et aussi ils ont utilisé des modulation d'ordre inférieure (QPSK et 4-QAM). De plus, les méthodes itératives basées sur l'estimation de canal peuvent mettre du temps à converger vers la solution optimale, ce qui ne répond pas aux exigences de communication V2X [10].
Pour résoudre ces problèmes, dans ce chapitre nous proposons un nouvel algorithme d'estimation de canal dédié aux communications véhiculaires. Les systèmes de communication V2X ciblées sont basés sur la forme d'onde OFDM, où une modulation d'ordre élevé est utilisée (64-QAM).

3.2 Modélisation d'un système de communication véhiculaire

Dans le cadre de nos travaux présentés dans ce chapitre, nous considérons une paire de communications émetteur/récepteur dans un réseau véhiculaire sans fil , dans lequel un nœud de communications 1 (véhicule connecté) transmet des informations à un autre nœud de communications 2 (véhicule connecté ou unité au bord de la route), comme illustré dans la figure ci-après :

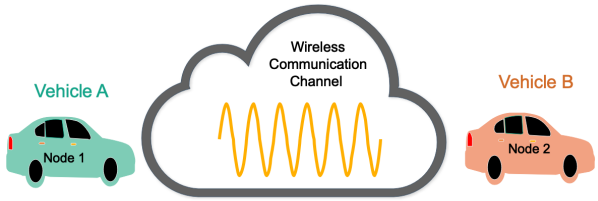


FIGURE 3.1: Modèle d'un système de communication véhiculaire.

Dans le cadre de notre travail, le réseau V2X proposé fonctionne selon la norme IEEE 802.11p, connue sous le nom WAVE aux États-Unis et ITS-G5 en Europe. La couche physique de cette norme est basée essentiellement sur la forme d'onde OFDM , qui consiste à diviser le signal transmis sur un grand nombre de sous-porteuses.
Le principe de fonctionnement d'un système de transmission basé sur le protocole IEE802.11p est représenté dans le graphe suivant :

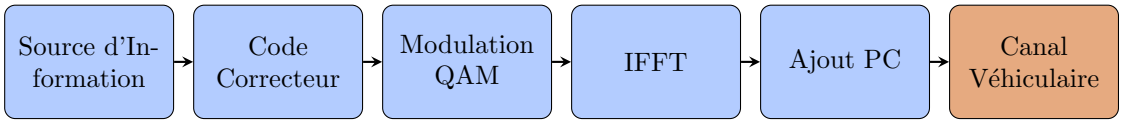


FIGURE 3.2: Émetteur IEEE 802.11p

Le système de transmission représenté à la Figure 3.2 comporte les blocs suivants :

— **Code Correcteur :** On utilise un code correcteur CRC, qui consiste à ajouter des bits de parité, qui seront utilisés pour la détection des bits erronés en réception.

— **Modulation QAM :** Ce bloc sert à transformer les données à transmettre en points de constellation en utilisant la modulation M-QAM (M est l'ordre de modulation).

— **IFFT :** Sert à transformer le signal à transmettre du domaine fréquentiel au domaine temporel.

— **Ajout PC :** On ajoute un Préfixe Cyclique (PC) après chaque symbole OFDM, pour éviter les interférences entre symboles OFDM.

Le signal issu de cet émetteur, qui sera envoyé à travers le canal véhiculaire est exprimé par l'équation suivante :

$$X(t) = \sum_{K=0}^{N-1} X_{(k)} e^{j2\pi kt/T},\ 0 \leq t < T \tag{3.1}$$

Où $X_{(k)}$ représente les symboles de données à transmettre, N est le nombre de sous-porteuses utilisées et T est la durée du symbole OFDM.
En revanche, la chaîne de réception est constituée comme illustrée au schéma blocs ci-après :

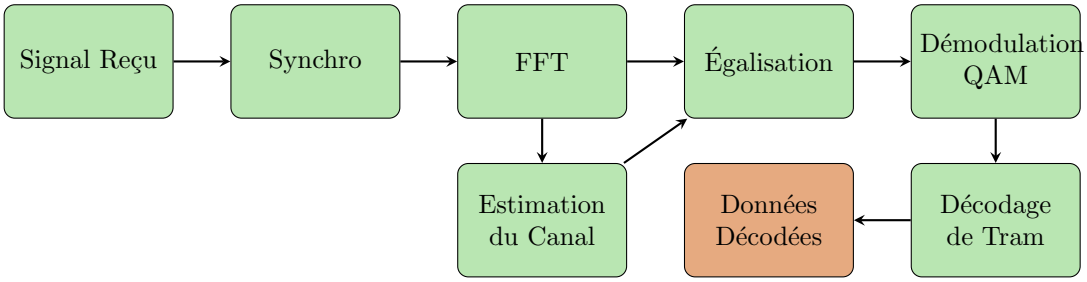


FIGURE 3.3: Récepteur IEEE 802.11p

Le récepteur représenté à la Figure 3.3 comporte les blocs suivants :

— **Synchronisation (Synchro) :**Ce bloc est utilisé pour la détection de la trame et l'alignement des symboles OFDM en utilisant les séquences d'apprentissages LTS et STS (Voir sous-section 2.6.1.2).

— **FFT :** Après la détection et l'alignement des symboles OFDM on transforme notre signal du domaine temporel au domaine fréquentiel, en outre le PC doit être supprimé avant d'appliquer la FFT.

— **Estimation du Canal :** Ce bloc sert à estimer les caractéristiques du canal afin de restituer notre signal transmis en utilisant les séquences d'apprentissages LTS.

— **Égalisation :** Ce bloc utilise les données issues de la sortie de la FFT et les valeurs CSI à la sortie de l'estimation pour restituer les données transmises.

— **Démodulation QAM :** Ce bloc sert à démapper les constellations des sous-porteuses bruyantes aux points de constellation idéaux les plus proches.

— **Décodage de Tram :** Ce bloc utilise les bits de parité pour détecter et corriger les bits erronés.

Le signal reçu à travers un canal sans fil véhiculaire dans le domaine fréquentiel peut être écrit comme suit :

$$Y_{(k)} = X_{(k)} H_{(k)} + W_{(k)} \tag{3.2}$$

Où $X_{(k)}$ est le signal transmis, $H_{(k)}$ désigne la réponse du canal sans fil, représentée par les valeurs CSI et $W_{(k)}$ est le bruit dans le canal transmission.
En outre l'équation 3.2, est exprimée sous la forme matricielle suivante :

$$\underset{[n*1]}{Y_k} = \underset{[n*n]}{X_k} \underset{[n*1]}{H_k} + \underset{[n*1]}{W_k} \tag{3.3}$$

Où, $\underset{[n*n]}{X_k} = Diag(X_{(k)})$ et n représente le nombre de sous-porteuses par symbole OFDM.

3.3 Canal de propagation véhiculaire

Le canal véhiculaire sans fil est donné comme un canal de propagation à trajets multiples avec un évanouissement doublement sélectif (double selective fading channel). Il est caractérisé par le retard (Delay path) et l'effet Doppler (Doppler shift) [3]. La réponse temporelle en bande de base du canal à trajets multiples est donnée par :

$$h(t,\tau) = \sum_{l=0}^{L-1} A_l(t)\delta(\tau - \tau_l(t)) \tag{3.4}$$

Où L, $A_l(t)$ et $\tau_l(t)$ représentent le nombre de trajets non nuls, les amplitudes complexes et le délai du retard pour chaque trajet l, respectivement.
Les paramètres $A_l(t)$ et $\tau_l(t)$ varient en fonction du temps, car le véhicule récepteur n'est pas fixe (en déplacement). Pour évaluer l'évolution de ces paramètres, on décrit le mouvement du véhicule par sa vitesse de déplacement ΔV, et l'angle d'arrivée θ_l par rapport au vecteur de la vitesse du déplacement du véhicule . [30](Voir Figure 3.4).

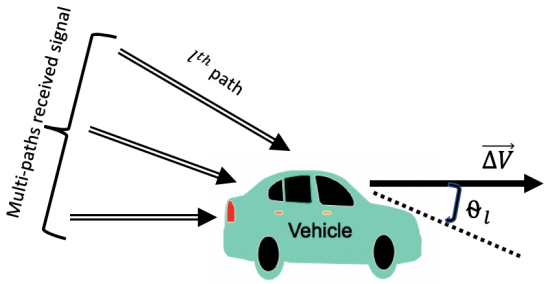


FIGURE 3.4: Signal à trajets multiples reçu dans un environnement véhiculaire.

Le retard de propagation relatif à chaque trajet l en fonction du temps peut s'exprimer comme suit :

$$\tau_l(t) = \tau_l(0) + \frac{\Delta V}{c} cos(\theta_l)t \tag{3.5}$$

Où c est la célérité (vitesse de la lumière) et ΔV est la différence de vitesse entre les deux véhicules communicants ($\Delta V = |V_1 - V_2|$).
La variation linéaire du retard entraîne une variation linéaire temporelle de la phase $\phi_l(t)$ du gain complexe $A_l(t)$. Cette phase est décrite en fonction du temps avec l'équation suivante :

$$\phi_l(t) = \phi_l(0) - 2\pi f_d cos(\theta_l)t \tag{3.6}$$

Dans les communications véhiculaires, la fréquence Doppler f_d peut être exprimée par la différence de vitesse entre les deux véhicules communicants ΔV comme indiqué ci-dessous :

$$f_d = \frac{\Delta V}{c} f_0 \tag{3.7}$$

Où c est la célérité (vitesse de la lumière) et f_0 est la fréquence centrale de communication.
En revanche, la propagation du signal sur ce canal peut induire sur chaque trajet un décalage Doppler, en plus du retard temporel. De ce fait au niveau du récepteur, on observe une superposition des différentes versions retardées et décalées par rapport au signal émis [3].
Les caractéristiques du canal de propagation véhiculaire sont supposées être statiques

sur un temps de cohérence T_c [82], qui est inversement proportionnel au décalage Doppler maximal f_d. Comme le montre l'équation suivante :

$$T_c \approx \frac{0.423}{f_d} \tag{3.8}$$

En fonction de la bande de cohérence du canal (inverse du temps de cohérence $1/T_c$), le canal est considéré comme étant un canal à évanouissement plat lorsque la bande de cohérence du canal est supérieure à la bande passante du signal. D'autre part, il s'agit d'un canal à évanouissement sélectif en fréquence lorsque la bande de cohérence du canal est inférieure à la bande passante du signal (entraîne des interférences intersymboles dans le domaine temporel).
Selon l'institut européen des normes de télécommunications (ETSI), le type d'environnement dans lequel les véhicules communiquent entre eux, a un impact majeur sur les messages échangés et donc sur le modèle de canal [23]. De ce fait, nous pouvons considérer 5 environnements véhiculaires majeurs : Rural avec line-of-sight (Rural LOS), Urbain avec une line-of-sight, et sans line-of-sight (Urban LOS et Urban NLOS) et Autoroute avec et sans line-of-sight (Highway LOS et Highway NLOS). Ces différents environnements se distinguent en fonction des différentes caractéristiques de modélisation du canal comme indiqué dans les tableaux 3.1,3.3 et 3.2 :

Paramètres	Rural LOS		
Délais de Retard (ns)	0	83	183
Gain moyen du trajet(dB)	0	-14	-17
Doppler Shifts	0	492	-295

TABLE 3.1: Paramètres de l'environnement Rural LOS

Paramètres	Urban LOS				Urban NLOS			
délais de Retard (ns)	0	117	183	333	0	267	400	533
Gain moyen du trajet (dB)	0	-8	-10	-15	0	-3	-5	-10
Doppler Shifts	0	236	-157	492	0	295	-98	591

TABLE 3.2: Paramètres des environnements Urban LOS et Urban NLOS

Paramètres	Highway LOS				Highway NLOS			
Délais de Retard (ns)	0	100	167	500	0	200	433	700
Gain moyen du trajet (dB)	0	-10	-15	-20	0	-2	-5	-7
Doppler Shifts	0	689	-492	886	0	689	-492	886

TABLE 3.3: Paramètres des environnements Highway LOS et Highway NLOS

3.3.1 Effet du canal sur les communications véhiculaires

L'effet à trajets multiples peut être visualisé comme étant un récepteur qui reçoit le signal de la part de plusieurs émetteurs groupés à la même position. Chaque émetteur envoie la même séquence de symboles avec des retards et des atténuations différents. Les échantillons reçus au niveau du récepteur n'appartiennent donc pas à un seul symbole, mais à la superposition de plusieurs symboles multiples avec des facteurs d'atténuation et des phases différents. Cela empêche la détection directe d'un symbole à partir d'un seul échantillon. Cette superposition de plusieurs symboles en un seul échantillon est appelée interférence entre symboles (ISI) . Les canaux ayant des ISI sont sélectifs en fréquence (Figure3.5) .

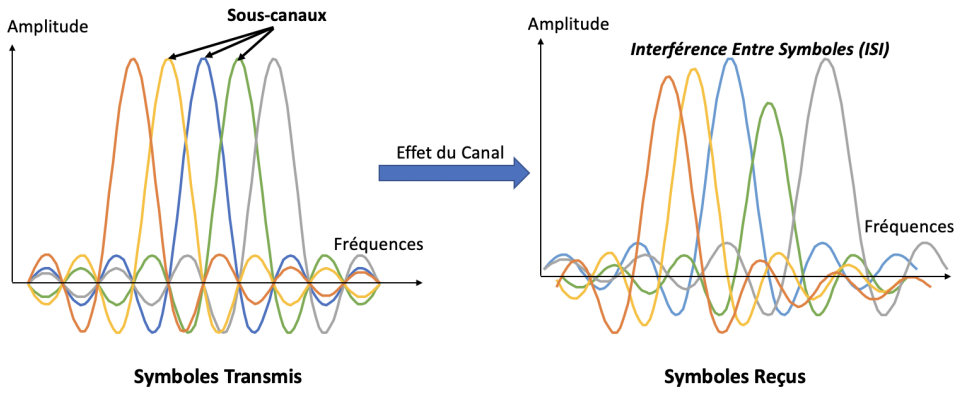


FIGURE 3.5: Canaux sélectifs en fréquence (Présence des ISI)

3.4 Techniques d'estimation de canal

L'estimation de canal est une étape indispensable dans la conception d'un récepteur dans les systèmes basés sur les formes d'ondes OFDM, tel que les systèmes de communication véhiculaires. Cependant, en raison de la variation temporelle et de la sélectivité en fréquence des canaux de transmission sans fil, l'estimation de canal reste un des challenges dans les systèmes de communication sans fil.
Par conséquent, Par conséquent, de nombreux travaux de recherche sur l'estimation de canal pour les systèmes OFDM ont été menés [34] [56]. En général, les méthodes d'estimation de canal peuvent être classifiées en trois catégories :

— Estimation de canal basée sur la réponse fréquentielle du canal (CFR).

— Estimation de canal basée sur un modèle paramétrique (PM).

— Estimation de canal itérative (ICE).

3.4.1 Estimation de canal basée sur la CFR

Les méthodes d'estimation de canal basées sur CFR, peuvent être considérées comme des approches de base traditionnelle. Elles consistent à utiliser des séquences d'apprentissages ou des symboles pilotes, de sorte que les symboles pilotes ou les séquences d'apprentissage sont d'abord estimés, puis les CFR des symboles de données peuvent être obtenues par interpolation ou par suivi basé sur le résultat obtenu.
Dans cette catégorie, on peut distinguer deux types d'algorithmes d'estimation de

canal, selon, si l'algorithme utilisé exige des informations statistiques sur la CFR ou non. Dans le cas où les connaissances statistiques sur la réponse du canal ne seraient pas exigées, l'algorithme de base le plus largement utilisé est le Least Square (LS) qui peut être formulé comme suit :

$$H_{LS} = \underset{H_{LS}}{Argmin} ||Y - X.H||_2^2 \tag{3.9}$$

où $\|.\|_2$, est la norme L_2, X est la séquence d'apprentissage, et Y désigne le vecteur d'observation correspondant. D'après [72], une expression plus optimisée de l'estimateur LS peut être écrite :

$$H_{LS} = Y.X^{-1} \tag{3.10}$$

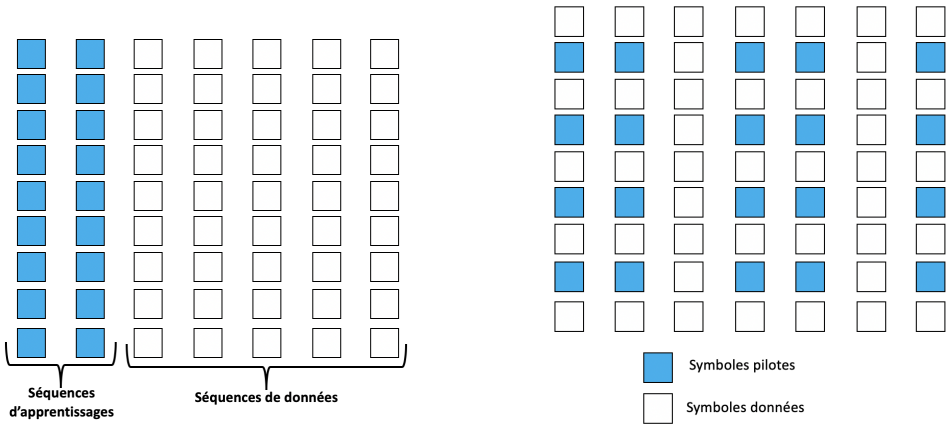
L'estimateur LS est adopté par plusieurs systèmes commerciaux et industriels en raison de sa simplicité [45].
Dans le cas où des connaissances statistiques sur le canal sont exigées, on utilise l'estimateur Minimum Mean Square Error (MMSE), qui consiste à exploiter les statistiques du second ordre des conditions de canal pour minimiser l'erreur quadratique moyenne (MSE), il suppose que la matrice de covariance du canal ($E(HH^H)$) et la variance du bruit (σ^2) sont connues [33].
L'estimateur MMSE peut être exprimé comme suit :

$$H_{MMSE} = R_H \left[R_H + \sigma^2 \left(\frac{1}{XX^H} \right) \right]^{-1} H_{LS} \tag{3.11}$$

Où R_H est la matrice de covariance du canal, σ^2 est la variance du bruit et H_{LS} désigne l'estimateur LS.
La complexité de l'estimateur MMSE est généralement plus importante car il nécessite d'effectuer une opération d'inversion d'une matrice.
Dans l'estimation de canal basée sur CFR, deux schémas d'allocations des pilotes sont envisagés : Decision-Directed Channel Estimation (DD-CE) et Pilot- Assisted Channel Estimation (PA-CE). Ces deux configurations d'allocations sont représentées à la Figure 3.6.
Dans DD-CE on utilise des séquences d'apprentissages placées avant les séquences de données. Donc en réception le processus d'estimation de canal consiste, en premier lieu à estimer la réponse fréquentielle des séquences d'apprentissage et ensuite utiliser cette CFR pour déterminer les séquences de bits des symboles de données [45].
Le problème principal de la configuration DD-CE est la propagation des erreurs, qui devient plus apparente avec l'augmentation de la fréquence Doppler [34]. Ce problème peut être résolu en insérant périodiquement les séquences d'apprentissage dans la trame transmise [71]. Dans ce cas, DD-CE est équivalent à PA-CE avec l'utilisation des symboles pilotes en bloc [17].
En revanche dans la configuration PA-CE on insère les symboles pilotes avec les symboles de données de manière séquentielle, comme indiquée à la Figure 3.6b. Dans la PA-CE les CSI des symboles pilotes sont d'abord estimés, puis les CSI correspondant aux symboles de données peuvent être obtenus en appliquant une fonction d'interpolation. Par conséquent, la précision des CSI des données symboles dépend de la précision de la fonction d'interpolation utilisée, ou l'utilisation d'interpolation d'ordre supérieur engendre plus de complexité de calcul.

3.4.2 Estimation de canal basée sur un modèle paramétrique

Dans l'estimation de canal basée sur un modèle paramétrique (PM), on estime les paramètres qui caractérisent la réponse de canal, comme : le nombre de trajets, les retards des trajets et le gain associé à chaque trajet, ce qui conduit à élaborer un modèle du canal.
Pour déterminer le nombre de trajets, différents critères de sélection sont abordés suivant le type de canal. Pour l'estimation des retards des trajets, différentes techniques



(a) DD-CE (b) PA-CE

FIGURE 3.6: Allocation des symboles pilotes et des séquences d'apprentissages.

de traitement de signal peuvent être adoptées [59]. La technique la plus utilisée est : Estimation of Signal Parameters via Rotational Invariance Techniques (ESPRIT). Avec l'estimation des retards des trajets, les gains peuvent être obtenus en utilisant des estimateurs linéaires typiques. La tache clé dans les approches basées sur PM est d'estimer les retards des trajets, en utilisant l'algorithme ESPRIT. La matrice de covariance du canal (R_H) dans le domaine fréquentiel peut être écrite sous la forme matricielle suivante :

$$R_H = E\left[HH^H\right] = VEV^H \tag{3.12}$$

Où E est une matrice diagonale qui contient les valeurs propres de R_H, dans un ordre décroissant et $V = [v(\tau_1), v(\tau_2), ..., v(\tau_l)]$ représente le vecteur de direction.
De manière générale la matrice de covariance R_H, est inconnue au récepteur, mais peut être estimée comme suit :

$$R_H \approx \frac{1}{N}\sum_{n=0}^{N-1} H_{LS}H_{LS}^H \tag{3.13}$$

Où H_{LS} représente l'estimateur LS de la CFR. Dans ce cas en appliquant une décomposition en valeurs propres (tel que SVD), R_H peut être factorisée comme suit :

$$R_H = [U_s, U_w]\begin{bmatrix}\Lambda_s & 0\\ 0 & \Lambda_w\end{bmatrix}\begin{bmatrix}U_s^H\\ U_w^H\end{bmatrix} \tag{3.14}$$

Où U_s et Λ_s sont la matrice des vecteurs propres et la matrice des valeurs propres correspondantes au sous-espace du signal, et U_w , Λ_w sont les matrices correspondant au sous-espace du bruit. Le bruit introduit par l'estimation LS, implique la présence du sous-espace de bruit.
Si on définit V1 et V2 comme des matrices de taille $(K-1)\times L$ construites respectivement par les premières $K-1$ lignes et les dernières $K-1$ lignes de V, où $V1 = [I_{k-1}0]V$ et $V2 = [0I_{k-1}]V$. D'après cette définition on peut écrire :

$$V2 = V1\Phi \tag{3.15}$$

où Φ est une matrice de rotation diagonale dont les éléments non nuls égale à $\Phi_{l,l} = e^{-j(2\pi\tau_l/T)}$.
L'estimation des retards des trajets peut être obtenue en obtenant les éléments de

Φ. Pour cela, on note, à partir de 3.13 et 3.14, que les colonnes de V et les colonnes de U_s couvrent le même espace de signal. Par conséquent, il existe une transformé linéaire inversible, qui peut mapper U_s à V.
Si nous définissons $U1$ et $U2$ de la même manière qu'on a défini $V1$ et $V2$ précédemment, de sort qu'ils soient des matrices de tailles $(K-1) \times L$ construites par les $K-1$ premières lignes et les dernières $K-1$ lignes de U_s, respectivement, on obtient :

$$\begin{aligned} V1 &= U1Q \\ V2 &= U2Q \end{aligned} \tag{3.16}$$

A partir de 3.15 et 3.16, on obtient :

$$U2 = U1\Psi \tag{3.17}$$

Ou, $\Psi = Q\Phi Q^{-1}$. Par conséquent les éléments de la matrice Φ sont les valeurs propres de Ψ, qui peuvent être estimées à partir de l'équation 3.17, comme suit :

$$\hat{\Psi} = (U1^H U1)^{-1} U1^H U2 \tag{3.18}$$

En revanche, on peut obtenir l'estimation des retards des trajets par :

$$\hat{\tau_l} = -\frac{T}{2\pi} Arg(\Phi_{l,l}) \tag{3.19}$$

Ou,$Arg(.)$, représente la phase des éléments complexes.
En général, l'approche basée sur PM peut améliorer les performances d'estimation de canal lorsque le nombre de paramètres inconnus est inférieur à la dimension de la matrice de covariance du canal [45]. C'est généralement le cas pour les canaux sans fil dans les zones rurales. Pour estimer directement les paramètres inconnus, il faut utiliser des techniques de traitement du signal basées sur les sous-espaces, comme MUSIC et ESPRIT [59], [63]. En outre, différents critères pourraient être adoptés pour déterminer le nombre de paramètres. Toutefois, les inconvénients de l'approche basée sur les PM sont : premièrement, on obtient une amélioration des performances lorsque le nombre de trajets multiples est très faible. Cependant, la plupart des communications ont lieu dans des zones urbaines, ou le nombre des trajets multiples est important. Cela limite l'utilisation de l'approche basée sur les PM. Deuxièmement, dans l'approche basée sur PM, la matrice de covariance requise par les techniques de traitement de signal basées sur un sous-espace est obtenue par la moyenne sur une longue séquence de symboles, cela rend l'approche basée sur la PM uniquement adaptée à la transmission continue [18] [45].

3.4.3 Estimation de canal itérative (ICE)

L'estimation de canal itérative par rapport à l'estimation de canal traditionnelle qui n'utilise que des pilotes ou des séquences d'apprentissage, utilise en plus des informations a priori sur les symboles de données qui sont fournis par le feedback du décodeur. La clé pour l'estimation de canal repose sur la méthode utilisée pour exploiter les informations a priori. De manière générale, les informations a priori peuvent être utilisées pour générer soit des symboles souples [61], [43], soit des symboles durs [11], [5]. De sort que les symboles souples représentent une prédiction du symbole transmis, qui peut être représenté comme suit :

$$\widetilde{X}_k = \sum_{i,j} P_{ij} Z_{ij} \tag{3.20}$$

Où i, j représentent les références de point constellation (par exemple pour une modulation QPSk $i = 0, 1 et j = 0, 1$), P_{ij} est l'information a priori obtenue à partir du

feedback du décodeur et Z_{ij} est le point de constellation correspondant.
D'autre part, un symbole dur, est généré en choisissant l'élément de la constellation qui correspond à la probabilité la plus élevée [11], [43], [5], il est décrit par :

$$\widetilde{X}_k = \underset{Z_{ij}}{argMax}(P_{ij}) \tag{3.21}$$

Dans l'approche itérative, il y a une estimation temporaire à chaque itération. Il est donc naturel d'envisager d'utiliser des estimations issues d'autres itérations si l'estimation actuelle n'est pas satisfaisante [42], [39]. Pour cela une fonction de coût ou de test de seuil peut être appliquée.
L'approche proposée dans [39] consiste à utiliser un seuil λ, pour obtenir les estimations de canal pour l'itération actuelle à partir d'autres itérations, qui est décrite par :

$$\widetilde{H}_k^i = \begin{cases} Y_k/\widetilde{X}_k^i & |\widetilde{X}_k^i| \geq \lambda \\ \widetilde{H}_k^{(i-1)} & |\widetilde{X}_k^i| < \lambda \end{cases} \tag{3.22}$$

Dans [39], une autre technique est proposée. Au lieu de choisir entre les deux termes dans 3.22, on utilise une somme pondérée des deux, avec un coefficient de pondération β, donnée par :

$$\widetilde{H}_k^i = \beta Y_k/\widetilde{X}_k^i + (1-\beta)\widetilde{H}_k^{(i-1)} \tag{3.23}$$

À noter que l'estimation de canal initiale ($\widetilde{H}_k^0$), est calculée en utilisant les symboles pilotes.
Dans la littérature il existe plusieurs algorithmes d'estimation de canal itérative tel que : Iteratively Reweighted Least Squares (IRLS) [83], l'algorithme Expectation-Maximization (EM) [92] [46] et l'algorithme Orthogonal Matching Pursuit (OMP) [36].
Il est généralement recommandé d'analyser le comportement de convergence d'un algorithme itérative. Par conséquent, de nombreux travaux utilisent Extrinsic Information Transfer (EXIT) pour déterminer cette convergence. Le diagramme EXIT est introduit dans [76] et [75] pour concevoir des constellations de signaux d'un schéma de démappage et de décodage itératif.

3.5 Algorithmes proposés

Dans cette section, nous proposons deux algorithmes pour estimer le canal de transmission dans un environnement véhiculaire :

— L'algorithme SCR (Sequential Constraint Removal).

— Et l'algorithme MSCR(Multiple Sequential Constraint Removal).

Dans ces algorithmes, on considère l'estimation du canal comme étant un problème inverse mathématique. Par conséquent, suivant l'équation 3.2 notre système peut être classifié dans les problèmes inverses discrets linéaires sous-déterminés (Underdetermined Linear Discrete Inverse Problems). Dans ce contexte, afin de pouvoir résoudre notre système d'équations en utilisant les algorithmes proposés (SCR et MSCR), on doit étendre la séquence d'apprentissage X_t et le signal reçu correspondant à cette séquence Y_t comme suit :

$$\underset{[(n+n)*n]}{X_{Ext}} = \begin{bmatrix} \underset{[n*n]}{X_t} \\ \\ \underset{[n*n]}{I} \end{bmatrix}, \underset{[(n+n)*1]}{Y_{Ext}} = \begin{bmatrix} \underset{[n*1]}{Y_t} \\ \\ \underset{[n*1]}{0} \end{bmatrix} \tag{3.24}$$

Où n représente le nombre des sous-porteuses OFDM, $\underset{[n*n]}{I}$ est la matrice identité de taille $[n*n]$ et $\underset{[n*1]}{0}$ est un vecteur nul de taille $(n*1)$.

3.5.1 Algorithme SCR

Cet algorithme est principalement basé sur un calcul itératif de l'estimateur LS, en se basant sur la séquence d'apprentissage étendu et le signal reçu correspondant à cette séquence étendue (Voir Eq 3.24).

Algorithm 1: Algorithme d'estimation de canal basé sur le SCR (Sequential Constraint Removal)

Input: Observation vector : Y_{Ext}
Input: Training sequence vector : X_{Ext}
Output: Estimated channel : H_{Est}
1 initialize : $H_{Int} = Y_{Ext}/X_{Ext}$, and $H_{New} = H_{Int}$
2 Set $H_{Old} = H_{New}$
3 Set $p =$ index position of the next $Max(H_{New})$
4 $X_{Ext}[(n+p), :] = 0$
5 Calculate : $H_{New} = Y_{Ext}/X_{New}$
6 **if** $|H_{New} - H_{Old}| \geq \epsilon$ **then**
7 Go to 2
8 **else**
9 End Iteration
10 Set $H_{Est} = H_{New}$
11 Return H_{Est}

Le processus d'estimation de canal à base de cet algorithme est constitué des étapes suivantes :

1. Calculer l'estimateur LS en utilisant les matrices étendues Y_{Ext} et X_{Ext} comme solution initiale H_{Int} (Équation 3.24). Ensuite, on met H_{Int} comme ancienne solution H_{Old}

2. Mettre H_{New} égal à H_{Old}.
3. définir p comme index de la valeur maximale dans H_{New}.
4. Mettre à zéro les valeurs de la ligne de la matrice X_{Ext}, qui correspond à la prochaine position de magnitude maximale dans $p + n$ (n est le nombre de sous-porteuses).
5. Calculer à nouveau l'estimateur LS H_{New} en utilisant la nouvelle séquence d'apprentissage X_{New}
6. Comparez H_{New} à H_{Old}.
7. Si la différence ($|H_{New}$ - $H_{Old}|$) est supérieur à ϵ, on répète le processus depuis l'étape 2.
8. Et si la différence entre H_{New} et H_{Old} est inférieur à ϵ, cela implique la fin d'itération.
9. Définir H_{New} comme solution de sortie H_{Est}.

3.5.2 Algorithme MSCR

Dans ce chapitre, nous proposons un nouvel algorithme d'estimation de canal : Multiple Sequential Constraint Removal (MSCR). cet algorithme comporte moins de complexité par rapport à l'algorithme SCR, où au lieu de mettre à zéro une ligne de la séquence d'apprentissage étendue (X_{Ext}) dans chaque itération, nous mettons plusieurs lignes à zéro en même temps (voir Algorithme 2).

Algorithm 2: Algorithme d'estimation de canal basé sur Multiple Sequential Constraint Removal (MSCR)

Input: Observation vector : Y_{Ext}
Input: Training sequence vector : X_{Ext}
Input: Number of Removed Sequence : $K = 52$
Output: Estimated channel : H_{Est}

```
Calculate : H_Int = Y_Ext / X_Ext
Sort = Sort descending [H_Int]
P = Position (Sort) in H_Int
for i = 1 to K do
    X_Ext[(n + p(i)), :] = 0
    X_New = X_Ext
Calculate : H_Est = Y_Ext / X_New
Return H_Est
```

Nous considérons comme entrée les matrices étendues Y_{Ext} et X_{Ext}, comme décrites précédemment dans l'équation 3.24.
Les principales étapes à suivre dans cet algorithme sont :

1. Calculer la solution initiale estimée en se basant sur le LS du système étendu H_{Int}.
2. Réorganiser les valeurs de la solution initiale H_{Int} en utilisant un tri décroissant.
3. Mettre à zéro les valeurs des lignes de la matrice X_{Ext}, correspondant à la position des premiers 52 valeurs dans le H_{Int} stocké.
4. Calculer l'estimateur LS en utilisant la nouvelle séquence d'apprentissage (H_{New}) obtenue à l'étape précédente,
5. Définir le résultat obtenu à l'étape précédente comme solution de sortie H_{Est}.

Une étude de convergence de l'algorithme MSCR a été réalisée afin de choisir le nombre de lignes optimal (K) à mettre à zéro . Pour cela, nous avons évalué le SER (Symbols Error Rate) par rapport à différents niveaux de SNR (rapport signal /

bruit) (entre $0dB$ et $20dB$), en utilisant diverses valeurs de K, de 2 jusqu'à 52 avec un pas de 10.
Les résultats présentés dans la Figure 3.7, montrent que la valeur optimale de convergence de l'algorithme est lorsque $K = 52$. Pour une valeur de K supérieur à 52 l'algorithme MSCR diverge (voir Figure 3.7). Cela est dû au fait que la matrice de séquence d'apprentissage X_{New} sera une matrice singulière, ce qui engendre plusieurs solutions divergentes en sortie.

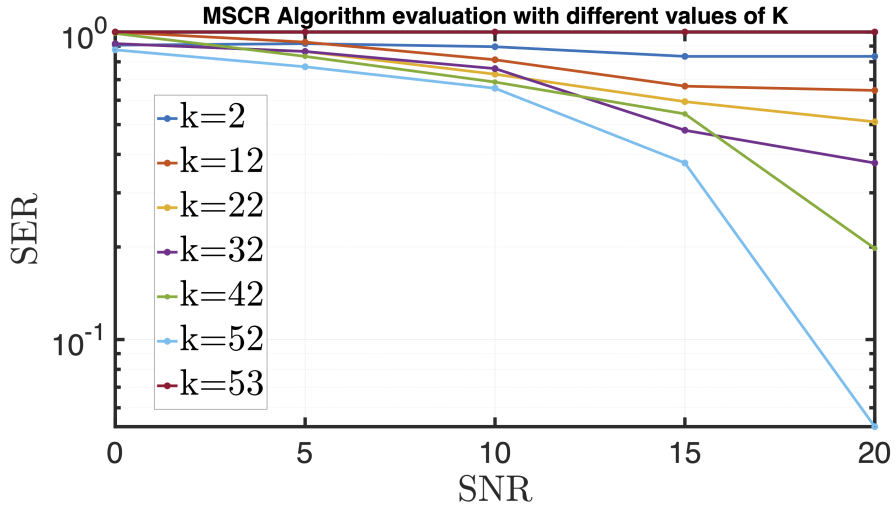


FIGURE 3.7: Évaluation de l'algorithme MSCR avec différentes valeurs K

3.6 Évaluation des performances

Afin de valider nos algorithmes proposés (MSCR et SCR) et de nous assurer que leurs performances répondent aux exigences environnementales des véhicules connectés, nous avons simulé une communication véhiculaire en utilisant Matlab. Cette simulation consiste à échanger des messages entre deux véhicules connectés dans les différents environnements véhiculaires.
Nous avons utilisé le framework V2VChannel de Matlab [3] pour modéliser le canal de transmission, qui a été réalisé sur la base des mesures réelles du retard, du gain et de la fréquence Doppler dans les différents types d'environnement véhiculaire, comme décrit dans la Section 3.3.
Nous avons réalisé 5 cas de figure de tests en fonction de type de l'environnement dans lequel les messages sont transmis comme suit : Rural-LOS, Urban-LOS, Urban-NLOS, Highway-LOS et Highway-NLOS. Les messages sont envoyés à une fréquence centrale de 5,9 GHz avec une bande passante de 10 Mhz. Les trames sont transmises en utilisant la forme d'onde OFDM avec une modulation QAM d'ordre 64 (64-QAM).

3.6.1 Évaluation du SER vs. SNR

Pour chaque test, nous avons évalué le taux d'erreur par symbole SER sur une plage de SNR comprise entre $0db$ et $20dB$. Les résultats ont été comparés avec les algorithmes de base utilisés (LS et MMSE).
Les résultats de tests pour l'environnement Rural-LOS présentés à la Figure 3.8. montre que l'algorithme MSCR a de meilleures performances comparant aux estimateurs LS, MMSE et aussi par rapport à l'algorithme SCR. Où l'algorithme MSCR

donne une bande inférieure du SER par rapport aux autres méthodes pour toutes les valeurs du SNR. Aussi, nous pouvons voir que l'algorithme MSCR proposé à un taux d'erreur par symbole SER comparable au SER obtenu dans un canal parfait.

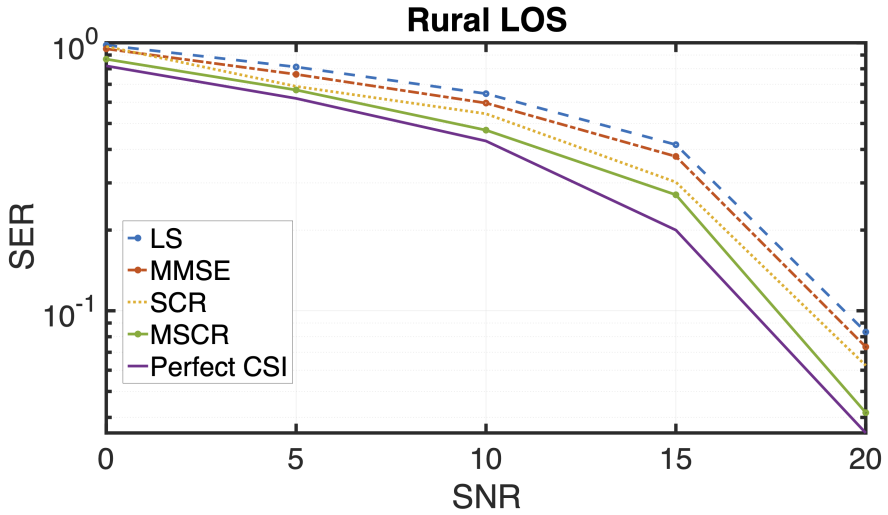


FIGURE 3.8: SER vs. SNR des algorithmes MSCR, SCR et des estimateurs MMSE, LS dans un environnement Rural-LOS.

La Figure 3.9 et la Figure 3.10 donnent respectivement les résultats de la simulation de l'environnement Highway-LOS et de l'environnement Highway-NLOS. Ils montrent que l'algorithme MSCR a des performances comparables avec les autres méthodes (les estimateurs LS MMSE et l'algorithme SCR) à des valeurs de SNR inférieure à $5dB$. Cependant, à des valeurs de SNR plus élevée (supérieure à $5dB$), l'algorithme MSCR donne de meilleures performances et peut atteindre un taux d'erreur par symbole (SER) autour de 7% pour une valeur de SNR égale à $20dB$.
De plus, nous pouvons voir sur la Figure 3.10 que dans l'environnement Highway-NLOS, les estimateurs LS et MMSE donnent moins de performances mais avec un pourcentage de SER acceptable (moins de 20%), où l'algorithme SCR a donné un SER important (plus de 50%) même avec des valeurs de SNR élevées.

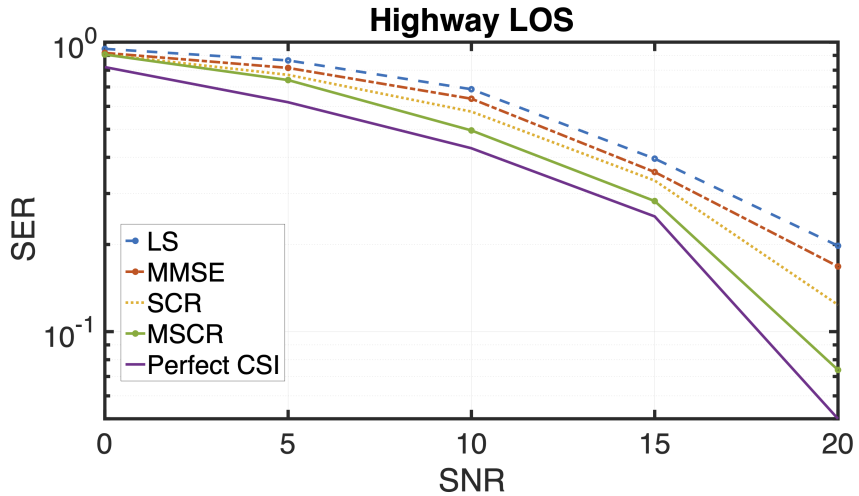


FIGURE 3.9: SER vs. SNR des algorithmes MSCR, SCR et des estimateurs MMSE, LS dans un environnement Highway-LOS.

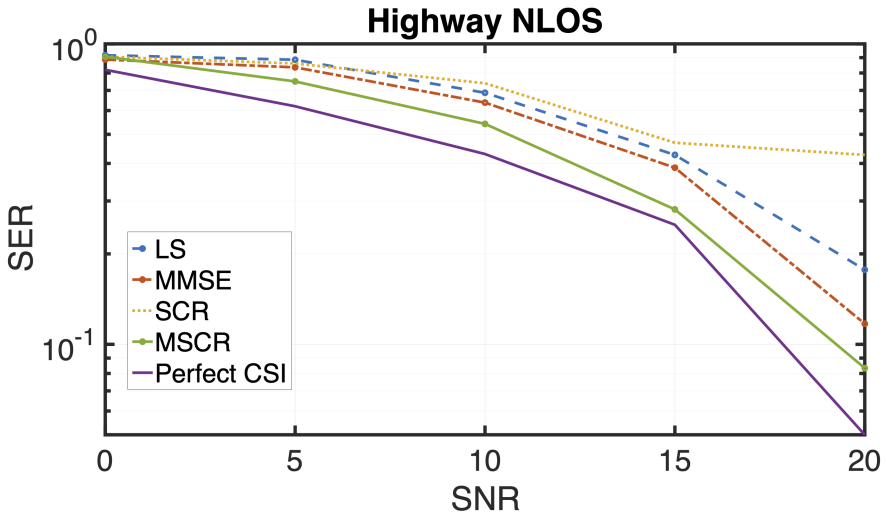


FIGURE 3.10: SER vs. SNR des algorithmes MSCR, SCR et des estimateurs MMSE, LS dans un environnement Highway-NLOS.

D'après la Figure 3.11, les résultats de l'environnement Urban-NLOS montrent que l'algorithme MSCR a de meilleures performances par rapport aux autres méthodes en termes de SER calculé pour toutes les valeurs SNR simulées (de $0dB$ à $20dB$). Il peut atteindre environ 10% de SER à une valeur de SNR de $20dB$. de plus ces performances sont comparables à celles d'un canal parfait.
La Figure 3.12 montre que dans l'environnement Urban-LOS, les estimateurs LS, MMSE et l'algorithme SCR ont des performances comparables à celles de l'algorithme MSCR dans de faibles valeurs de SNR (de $0dB$ à $10dB$). Cependant, pour des valeurs de SNR plus élevées, l'algorithme MSCR donne des performances meilleures (environ 5% SER), où nous pouvons voir les estimateurs LS et MMSE donnent un pourcentage SER de 25% et 23% respectivement. L'algorithme SCR donne de meilleures performances avec un SER égale à 18%, mais toujours moins que le SER obtenue par l'algorithme MSCR qui est de l'ordre de 5%.

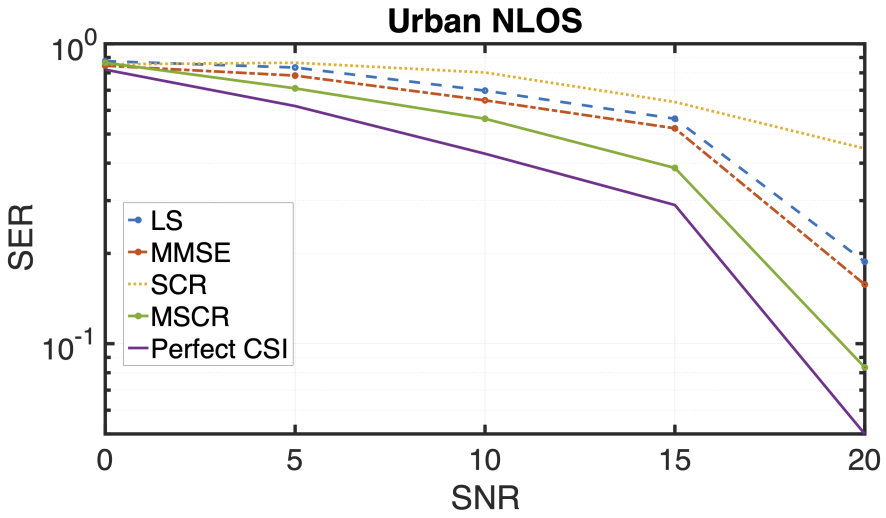


FIGURE 3.11: SER vs. SNR des algorithmes MSCR, SCR et des estimateurs MMSE, LS dans un environnement Urban NLOS

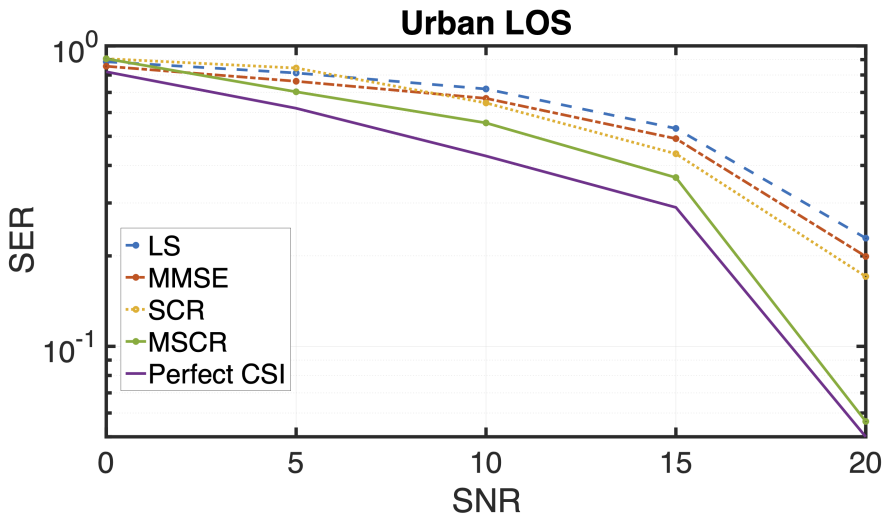


FIGURE 3.12: SER vs. SNR des algorithmes MSCR, SCR et des estimateurs MMSE, LS dans un environnement Urban LOS

Étant donné que notre algorithme d'estimation de canal proposé (MSCR) est destiné à être utilisé dans un environnement véhiculaire qui a des exigence critiques, il doit respecter un faible taux d'erreur de symbole (SER) dans chaque scénario. Selon les résultats de simulation, l'algorithme SCR n'est pas adapté à des environnements avec une configuration NLOS car il donne un SER supérieur à 50%. De plus, l'algorithme SCR n'est pas toujours fiable car il a nécessité plusieurs itérations et une tolérance d'erreur ajustable (*epsilon*) pour converger vers la solution satisfaisante. En revanche, notre algorithme proposé MSCR donne des performances plus élevées en termes de SER calculé dans une plage de SNR allant de $0dB$ à $20dB$ dans tous les types d'environnement. De plus, il offre des performances proches de celles obtenues par un canal parfait. De ce fait, il est plus adapté à tous les scénarios d'environnement véhiculaire car il converge à chaque fois vers la solution optimale.

3.6.2 Temps de latence (temps d'exécution)

Puisque l'environnement véhiculaire nécessitait une réponse en temps réel, nous avons évalué le temps d'exécution moyen des algorithmes MSCR et SCR. Les résultats sont présentés au Tableau suivant :

Algorithmes	MSCR	SCR
Temps d'exécution moyen (ms)	3.8	17.5

TABLE 3.4: Temps d'exécution moyen des algorithmes MSCR et SCR

D'après les résultats obtenus concernant le temps d'exécution de l'algorithme MSCR et de l'algorithme SCR (Tableau 3.4), nous avons constaté que les deux algorithmes satisfont le temps de latence minimale requis pour les communications V2X, selon [10]. Cependant, nous pouvons voir que l'algorithme MSCR atteint $3,8ms$, ce qui est 4 fois plus rapide que l'algorithme SCR ($17,5ms$). Cela est dû au fait que le temps de latence de l'algorithme SCR dépend du nombre d'itérations nécessaires pour converger vers la solution optimale.

3.6.3 Intervalle de confiance (CI)

Pour valider la précision de nos résultats de simulation à l'aide de l'algorithme proposé MSCR, nous avons calculé l'intervalle de confiance (CI) du SER en fonction de plusieurs simulations effectuées, avec les valeurs SNR : $15dB$ et $20dB$. Le niveau de confiance le plus couramment utilisé est de 95%, où un intervalle de confiance de 95% est une plage de valeurs, dans laquelle nous pouvons être sûrs à 95% qu'il contient la vraie moyenne de SER.
L'intervalle de confiance est calculé comme suit :

$$CI = \bar{X} \pm Z \times \frac{\sigma}{\sqrt{N}} \tag{3.25}$$

Où Z est le score correspondant pour le niveau de confiance choisi (pour un niveau de confiance de 95% Z est égal à $1,96$), $\bar{X}$ est la moyenne des SER, σ est l'écart-type, et N est le nombre de simulations effectuées.

Nombre d'exécutions (N)	Intervalle de Confiance
1×10^3	0.3592 ± 0.0047
5×10^3	0.3575 ± 0.0020
1×10^4	0.3607 ± 0.0017
1×10^5	0.3589 ± 0.00040

TABLE 3.5: Intervalle de confiance à 15 dB de SNR

Nombre d'exécutions (N)	Intervalle de Confiance
1×10^3	0.1374 ± 0.0043
5×10^3	0.1376 ± 0.0019
1×10^4	0.1369 ± 0.0013
1×10^5	0.1379 ± 0.00043

TABLE 3.6: Intervalle de confiance à 20 dB de SNR

L'intervalle de confiance a été calculé en fonction de plusieurs nombres d'exécutions de la simulation (1×10^3, 5×10^3, 1×10^4 et 1×10^5), où les différents types d'environnements véhiculaires ont été pris en compte dans ces simulations.
L'intervalle de confiance (au niveau de 95%) pour chaque nombre de simulations, à $15dB$ et $20dB$ de SNR, est présenté respectivement dans les tableaux 3.5 et 3.6. Là où nous pouvons voir que, lorsque le nombre de simulations est grand, l'intervalle de confiance devient assez étroit, ce qui signifie qu'il a beaucoup plus de précision que lorsque nous avons un nombre d'exécutions de simulations plus petit.

3.7 Conclusion

Dans ce chapitre, nous avons présenté une nouvelle approche d'estimation de canal dédiée au système de communication véhiculaire basé sur l'OFDM. La méthode proposée est basée sur l'utilisation de l'algorithme MSCR qui utilise la séquence d'apprentissage connue pour estimer les CSI. Cet algorithme a été évalué sur un canal doublement sélectif modélisé sur la base des caractéristiques environnementales des canaux véhiculaires.
Les résultats de la simulation montrent que l'algorithme MSCR peut atteindre un faible taux d'erreur par symbole même dans des conditions d'environnement véhiculaires très difficiles, avec un bon temps de latence. Ainsi, l'approche proposée avec le nouvel algorithme MSCR convient aux exigences des communications V2X.

Chapitre 4

Identification de l'environnement véhiculaire

Sommaire

4.1 Introduction 60
4.2 Machine learning 61
4.2.1 Modèles de Machine Learning 61
4.2.1.1 Support Vector Machine (SVM) 61
4.2.1.2 K-Nearest Neighbors (KNN) 61
4.2.1.3 Gaussian Naive Bayes (GNB) 62
4.2.1.4 Random Forest (RF) 62
4.3 Deep Learning 63
4.3.1 Architectures de deep learning 63
4.3.1.1 Artificial neural networks (ANN) 63
4.3.1.2 Convolutional Neural Networks (CNN) 64
4.4 Méthodologie 66
4.4.1 Architecture proposée 67
4.4.1.1 Architecture CNN 67
4.4.1.2 Architecture ANN 68
4.4.2 Génération de Data-Set 68
4.5 Évaluation des performances et résultats 70
4.5.1 Évaluation des performances de l'approche basée sur LTS 70
4.5.2 Évaluation des performances de l'approche basée sur CSI 75
4.5.3 Comparaison des performances de notre modèle CNN avec des architectures existantes 79
4.5.4 Fiabilité de performance minimaux 82
4.6 Conclusion 82

4.1 Introduction

L'intelligence artificielle a été établie comme un acteur majeur dans le développement des systèmes de transports intelligents, pour assurer la sécurité routière. Par conséquent, les futurs véhicules autonomes connectés devraient pouvoir être conscients des conditions environnementales autour. Donc ils doivent être équipés d'un système intelligent capable d'identifier le type d'environnement dans lequel le véhicule roule, pour permettre aux véhicules de prendre la bonne décision de conduite autonome, en fonction de l'environnement aperçu [15] [55].
Dans la littérature, de nombreux travaux de recherche se sont concentrés sur la perception de l'environnement basée sur Deep Learning afin de prendre des décisions critiques telles que la vitesse du véhicule pour adopter le bon comportement de conduite des voitures autonomes. Dans [89], les auteurs proposent une nouvelle méthode appelée approche de perception intégrée pour construire l'environnement. Où ils utilisent les données routières, telles que les distances par rapport aux marques de voies environnantes, fournies à partir d'images vidéo. Ces données sont utilisées comme caractéristiques d'entrée d'un modèle de réseau de neurones afin de prendre les bonnes décisions de conduite.
Une approche d'identification de l'environnement routier est présentée dans [65]. Ils utilisent ainsi les données vidéo de la zone d'autoroute enregistrées dans diverses conditions météorologiques Afin de développer un système de vision pour reconnaître les limites de la zone d'autoroute et mettre à jour le véhicule aux conditions de conduite sur l'autoroute. Dans [62], une nouvelle méthode de perception des environnements urbains est présentée. Cette approche est basée sur l'utilisation des images vidéo fournies par une caméra embarquée dans un véhicule pour entraîner un réseau de neurones, afin d'obtenir un modèle de navigation conditionnelle qui permet de prendre des commandes directionnelles de haut niveau au préalable.
Une méthode de perception de l'environnement urbain véhiculaire pour les véhicules autonomes est présentée dans [38]. cette approche consiste en un algorithme de fusion de données GPS, radar et Lidar afin de prendre des décisions de conduite tout en assurant la sécurité du conducteur. Une approche de perception de l'environnement est établie dans [15], qui vise à réaliser un véhicule autonome en zone urbaine. Dans cette méthode, les auteurs utilisent un LIDAR rotatif à 64 faisceaux avec un algorithme spécifique non supervisé, puis génèrent des cartes haute résolution de l'environnement, qui permettent au véhicule d'activer les paramètres de conduite appropriés pour cet environnement. Les auteurs de [78], établissent une approche basée sur l'utilisation de la fusion de données comprenant des capteurs caméra, LiDAR à 360 degrés et GPS/IMU déployés dans un véhicule, afin d'obtenir une présentation des environnements. Ainsi, le véhicule peut prendre des décisions correctes et autonomes en fonction de l'environnement dans lequel il roule.
Presque toutes les approches présentées précédemment reposent essentiellement sur l'utilisation de capteurs spécifiques tels que les caméras, les radars et les LIDAR. La collecte de données basée sur ces capteurs nécessite une quantité importante de ressources informatiques et énergétiques [91].
Pour éviter ces difficultés, nous proposons une nouvelle approche d'identification de l'environnement basée sur le Deep Learning dédié aux véhicules autonomes sans nécessiter de capteurs spécifiques. Pour cela, nous exploitons les caractéristiques du canal sans fil partagé, entre les véhicules communicants dans un réseau véhiculaire.
Puisque les valeurs CSI sont la représentation la plus précise des caractéristiques du canal sans fil [58], nous utilisons les valeurs CSI estimées à partir des paquets échangés entre les véhicules via des communications V2V, comme caractéristiques d'entrée pour notre modèle de réseau de neurones convolutif proposé. Ensuite, ce modèle est capable d'identifier de manière fiable l'environnement dans lequel le véhicule roule. Ainsi, le véhicule peut définir les bons paramètres de conduite autonome (tels que les limites de vitesse) correspondant à l'environnement identifié.

4.2 Machine learning

Le terme **"Machine Learning"** a été inventé par Arthur Samuel en 1959, un pionnier américain dans le domaine du jeu vidéo et de l'intelligence artificielle et a déclaré que «cela donne aux ordinateurs la capacité d'apprendre sans être explicitement programmés». Et en 1997, Tom Mitchell a donné une définition mathématique et relationnelle, selon laquelle «on dit qu'un programme informatique apprend de l'expérience E par rapport à une tâche T et à une mesure de performance P, si sa performance sur T, telle que mesurée par P, s'améliore avec l'expérience E.

D'un autre côté, Machine Learning est une sous-catégorie de AI, qui consiste à étudier des algorithmes qui reposent sur des modèles mathématiques basés sur des échantillons de données, appelés "Data-Set", afin de permettre à un ordinateur de faire des prédictions ou des décisions sans être explicitement programmé pour le faire [26] [50]. Les algorithmes de Machine Learning sont utilisés dans divers domaines tels que : diagnostic dans le domaine médical, vision par ordinateur et la conduite autonome des véhicules intelligentes..etc.
Les solutions basées sur les algorithmes de Machine Learning peuvent être classées en trois catégories, en fonction de la nature de la sortie souhaitée d'un système, comme suit :

- **Classification :** Le problème est considéré comme étant un problème de classification lorsque les entrées sont divisées en deux classes ou plus, et que le système doit produire un modèle qui attribue des entrées invisibles à une ou plusieurs classes (classification multi-étiquettes). Ceci est généralement abordé de manière supervisée. Le filtrage anti-spam est un exemple de classification, où les entrées sont des e-mails (ou autres) et les classes sont «spam» et «non spam».
- **Régression :** il est également considéré comme un problème supervisé. Ou dans cette catégorie les sorties du modèle de prédiction abordé sont de forme continue plutôt que discrète. Comme exemple sur cette catégorie on a : la prédiction de la température dans la journée.
- **Clustering :** Lorsqu'un ensemble d'entrées doit être divisé en groupes. Contrairement à la classification, les groupes ne sont pas connus à l'avance, ce qui en fait généralement une tâche non supervisée. cette catégorie est utilisée dans l'analyse des données des réseaux sociaux.

4.2.1 Modèles de Machine Learning

4.2.1.1 Support Vector Machine (SVM)

Support Vector Machine (SVM) est un modèle de Machine Learning supervisé qui est considéré comme une approche de classification, mais peut être utilisé dans les deux cas de problème soit la classification ou bien la régression. Il peut facilement gérer plusieurs types de variables continues et catégorielles. SVM consiste à construire un hyperplan dans un espace multidimensionnel pour séparer différentes classes. SVM génère un hyperplan optimal de manière itérative, qui est utilisé pour minimiser une erreur. L'idée de base de SVM est de trouver un hyperplan marginal maximal (MMH) qui divise au mieux l'ensemble de données en classes.

4.2.1.2 K-Nearest Neighbors (KNN)

L'algorithme KNN est un modèle de Machine Learning destiné à résoudre les problèmes de classification et aussi peut être utilisé pour des problèmes de régression. il est basé essentiellement, sur une approche de similarité des fonctionnalités. K-NN est un algorithme d'apprentissage dit non-paramétrique et paresseux. Non-paramétrique, car il n'y a aucune hypothèse pour la distribution des données. En d'autres termes, la

structure du modèle est déterminée à partir de l'ensemble de données. Cela sera très utile dans la pratique où la plupart des ensembles de données dans le monde réel ne suivent pas des hypothèses théoriques mathématiques. L'algorithme est dit paresseux parce qu'il n'a besoin d'aucun point de données d'entraînement pour la génération du modèle. Toutes les données d'entraînement sont utilisées dans la phase de test. Cela rend la phase de test plus lente et plus coûteuse, ça veut dire elle nécessite plus de temps et de mémoire.

Dans KNN, K est le nombre de voisins les plus proches. Le nombre de voisins est le principal facteur décisif. Supposons que P soit le point pour lequel on doit prédire l'étiquette. Tout d'abord, on cherche les K points les plus proches de P, puis on classe P par le vote majoritaire de ses K voisins. Chaque objet vote pour sa classe et la classe avec le plus de votes est prise comme prédiction. Pour trouver les points similaires les plus proches, on cherche généralement la distance entre les points à l'aide de mesures de distance telles que la distance euclidienne et la distance de Hamming. de manière générale K-NN consiste à suivre les étapes suivantes : Calculer la distance, trouver les voisins les plus proches et voter pour les labels.

4.2.1.3 Gaussian Naive Bayes (GNB)

GNB est une technique simple des modèles de Machine Learning dédiée aux problèmes de classifications, elle fait partie d'une famille d'algorithmes nommés classificateurs naïfs de Bayes, qui sont basés sur un principe que la valeur d'une caractéristique particulière est indépendante de la valeur de toute autre caractéristique, quelle que soit la variable de classe.
Par exemple, un fruit peut être considéré comme une pomme s'il est rouge, rond et d'environ 10 cm de diamètre. Un classificateur naïf de Bayes considère que chacune de ces caractéristiques contribue indépendamment à la probabilité que ce fruit soit une pomme, quelles que soient les corrélations possibles entre les caractéristiques de couleur, de rondeur et de diamètre.
Pour certains types de modèles de probabilités, les classificateurs Bayes naïfs peuvent être entraînés très efficacement dans un cadre d'apprentissage supervisé. Dans de nombreuses applications pratiques, l'estimation des paramètres des modèles Bayes naïfs utilise la méthode du maximum de vraisemblance ; en d'autres termes, on peut travailler avec le modèle naïf de Bayes sans accepter la probabilité bayésienne ou utiliser des méthodes bayésiennes.

4.2.1.4 Random Forest (RF)

Random Forest ou les forêts de décision aléatoires sont une méthode d'apprentissage appliquée pour les problèmes de classification et de régression, qui fonctionnent en construisant une multitude d'arbres de décision au moment de l'apprentissage et en produisant la classe qui est le mode des classes (classification) ou la prédiction moyenne (régression) des arbres individuels.
Le modèle RF se compose d'un grand nombre d'arbres de décision individuels qui fonctionnent comme un ensemble. Chaque arbre individuel dans la forêt aléatoire donne une prédiction de classe et la classe avec le plus de votes devient la prédiction de notre modèle.
Le concept fondamental de la forêt aléatoire est simple mais puissant. La faible corrélation entre les modèles (arbres) est la clé. Ces modèles non corrélés peuvent produire des prédictions d'ensemble plus précises que n'importe quelles prévisions individuelles. La raison est que les arbres se protègent les uns les autres de leurs erreurs individuelles (tant qu'ils ne se trompent pas tous constamment dans la même direction). Alors que certains arbres peuvent faire une prédiction fausse, de nombreux autres arbres auront une prédiction correcte, de sorte qu'en tant que groupe, les arbres peuvent avoir la bonne prédiction.

4.3 Deep Learning

Deep learning (également connu sous le nom d'apprentissage structuré profond) fait partie d'une famille de méthodes de machine learning basées sur des réseaux de neurones artificiels avec apprentissage par représentation. L'apprentissage peut être supervisé, semi-supervisé ou non supervisé. Le traitement de l'information et les nœuds de communications distribués dans les réseaux de neurones artificiels, ont été du fonctionnement du système nerveux du cerveau humain.
Diverses architectures de deep learning telles que deep neural networks (DNN), deep belief networks(DBN), recurrent neural networks (RNN) and convolutional neural networks (CNN) ont été appliquées dans plusieurs domaines, dont on cite : la vision par ordinateur, la reconnaissance vocale, le traitement du langage naturel, la reconnaissance faciale, le filtrage des réseaux sociaux, la traduction automatique, la bio-informatique , la conception de médicaments et l'analyse d'images médicales.

4.3.1 Architectures de deep learning

Ils existent plusieurs types d'architecture de deep learning, dans le cadre de notre travail, nous avons considéré deux types qui sont :
— Artificial neural networks (ANN)
— Convolutional Neural Networks (CNN)

4.3.1.1 Artificial neural networks (ANN)

Les réseaux de neurones artificiels (ANN), sont inspirés du fonctionnement du système cérébral humain, ils sont basés sur un ensemble d'unités appelé neurones qui sont connectés entre eux pour traiter et envoyer les informations.

L'élément de base pour Un réseau ANN est le neurone, qui se compose essentiellement des entrées $x = (x_1, x_2, .., x_n)$ d'un biais b et de la sortie $h_{w,b}(x)$ comme illustré dans la Figure suivante :

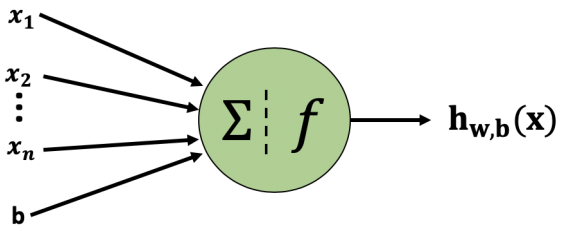


FIGURE 4.1: neurone

Toutes les entrées sont connectées au neurone à travers un synapse, ou chaque synapse comporte un poids W
la sortie du neurone peut s'exprimer comme suit :

$$h_{W,b}(x) = f(\sum_{i=1}^{n} W_i x_i + b) \tag{4.1}$$

W_i est le poids associé à l'entrée x_i et b est le bais. f représente la fonction d'activation, D'où, il existe plusieurs types de fonctions d'activation utilisées dans les réseaux de neurones. Les fonctions les plus utilisées sont : la fonction **sigmoïde**, la fonction **Tanh** et la fonction Rectifier Linear Unit (**ReLu**).
Afin de pouvoir élaborer un réseau de neurones à plusieurs couches, on peut relier plusieurs neurones entre eux, de sorte que la sortie des neurones d'une couche représente les entrées des neurones dans la couche suivante (voir Figure 4.2)

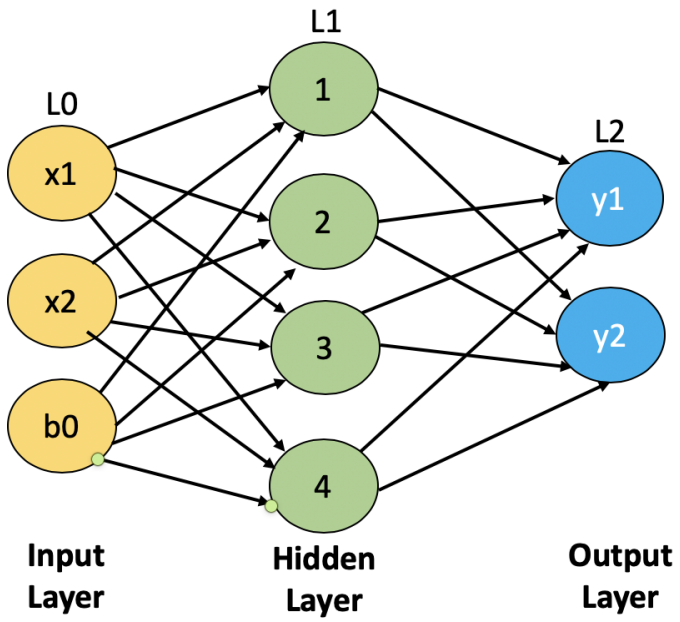


FIGURE 4.2: réseaux de neurones artificiels (ANN)

Dans le réseau de neurones établi à la Figure 4.2 on a 3 couches :

— Une couche d'entrée $L0$ qui est constitué des entrées $x1$, $x2$ et d'un bais $b0$.
— Une couche de sortie $L2$ qui comporte 2 sorties $y1$ et $y2$.
— Entre les deux couches précédentes, on a une couche cachée $L1$, qui contient 4 neurones.

4.3.1.2 Convolutional Neural Networks (CNN)

Les réseaux de neurones convolutifs (CNN ou ConvNet) font partie d'une classe de Deep Learning, appliqués essentiellement à l'analyse et le traitement des données à 2 dimensions ou plus comme les images.
Le nom de «réseau neuronal convolutif» vient du fait que les couches employées dans ces réseaux de neurones appliquent une opération de convolution mathématique.
Dans les réseaux CNN les caractéristiques importantes ne sont pas extraites d'une seule entrée mais de son voisinage aussi en utilisant des filtres de convolution. Les convolutions ajoutées agissent comme des extracteurs de caractéristiques et le fait qu'elles soient entraînées en fait automatiquement l'extraction des caractéristiques. Ces caractéristiques peuvent ensuite être classées ou traitées en utilisant des réseaux ANN afin de produire les sorties souhaitées. Un réseau de neurones convolutif CNN inclut les types de couches suivantes :

1. **La couche convolutive :** cette couche est la spécificité des réseaux CNN. Elle prend des matrices de données comme entrée, qui comprend généralement 3 dimensions . Ces matrices sont également appelées *tensors* dans certaines bibliothèques telles que Tensorflow. En outre, une quatrième dimension pourrait être ajoutée à ces matrices afin de traiter plusieurs entrées en même temps. Cette pratique est appelée batching et elle est largement utilisée dans les accélérateurs matériels. Par exemple une image en entrée d'un réseau CNN peut être envisagée comme étant une matrice de dimensions $W \times H \times C$. Où, la valeur de C représente le nombre de canaux dans l'image d'entrée (3 dans le cas de RVB et 1 dans le cas d'images en niveaux de gris). W et H représentent les dimensions de l'image, à savoir la largeur et la hauteur. Les

neurones de ces couches convolutives sont des filtres. Dans les réseaux ANN chaque neurone prend chaque entrée et la multiplie par un poids. Une couche convolutive fait la même chose avec deux modifications. Le premier étant, les entrées du neurone ne sont pas des valeurs individuelles des matrices de données. La deuxième modification est le partage du poids. Le neurone de couche convolutive multiplierait chaque fenêtre d'entrée avec le même filtre, ce qui réduit considérablement le nombre de paramètres. L'opération de convolution dans ces couches est représentée à la Figure 4.3. A la sortie de la couche de convolution on applique une fonction d'activation (ReLU, Tanh,Sigmoid).

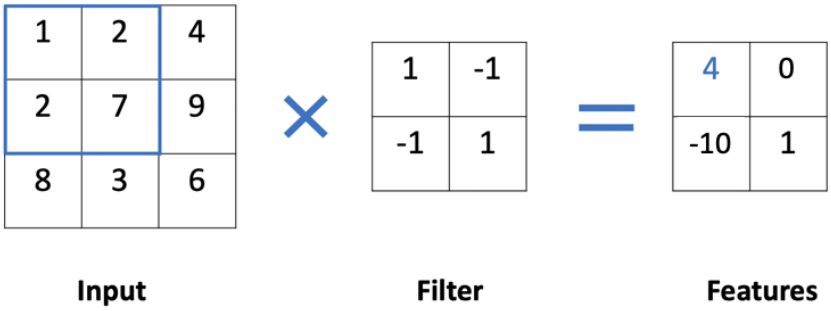


FIGURE 4.3: Opération de convolution

2. **La couche Pool :** Cette couche est périodiquement insérée dans les CNNs et sa fonction principale est de réduire la taille du volume, ce qui rend le calcul rapide, réduit la mémoire et empêche également le overfitting. Deux catégories de ce type de couches sont largement utilisées : **Max Pooling** et **Average Pooling**.
3. **la Couche entièrement connectée :** cette couche est une couche de réseau ANN, qui prend comme entrée les sorties de la couche précédente et comporte une sortie à 1 dimension.

4.4 Méthodologie

Dans le cadre de notre travail, nous considérons le processus d'identification de l'environnement véhiculaire comme un problème de classifications multi-classes, ainsi nous proposons deux méthodes pour y faire face :

— La première méthode est basée sur l'utilisation des longues séquences d'apprentissage (LTS) de la trame reçue

— Et la deuxième approche est basée sur l'utilisation des valeurs CSI estimées.

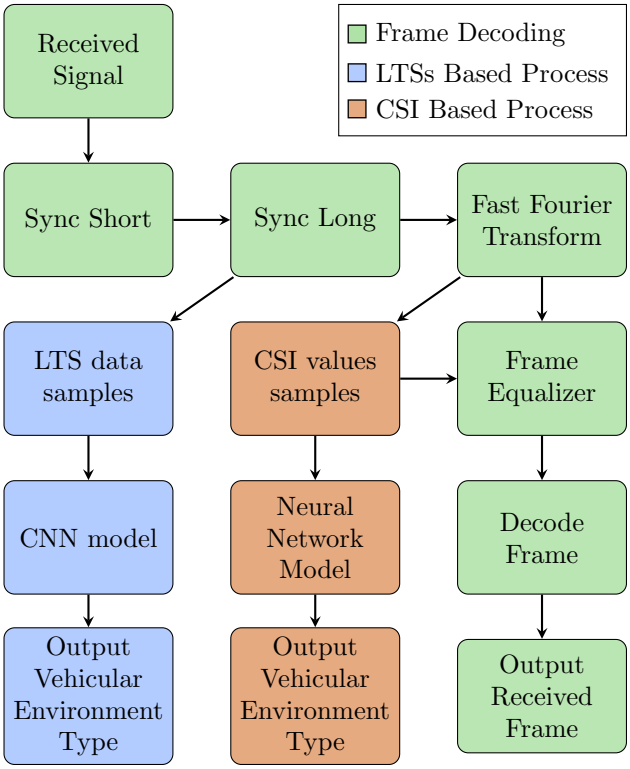


FIGURE 4.4: Organigramme décrivant le processus d'identification de l'environnement véhiculaire

Le processus d'identification de l'environnement pour les deux approches indiquées précédemment (basé sur LTS ou CSI), est représenté sur le schéma de la Figue 4.4. Il comporte les étapes suivantes :

1. collection des valeurs LTS ou CSI qui comprennent 128 échantillons de données.

2. Ces 128 échantillons sont utilisés comme entrée pour le modèle de réseau de neurones.

3. En sortie de réseau de neurones notre modèle est capable de nous fournir le type d'environnement dans lequel la voiture connectée circule.

4.4.1 Architecture proposée

Pour atteindre notre objectif qui est l'identification de l'environnement véhiculaire, nous avons proposé deux architectures de réseaux de neurones :

— Une architecture CNN comme le montre la Figure 4.5.

— ET une architecture CNN (voir Figure 4.6).

4.4.1.1 Architecture CNN

Le modèle CNN proposé est constitué comme suit : premièrement, il commence par deux couches convolutives similaires à 1 dimension (1D), où ces deux couches comprennent 45 filtres. Ensuite, nous avons une troisième couche convolutive à 1D, comprenant 20 filtres. Ceci est suivi par deux autres couches convolutives à 1D, qui comprennent respectivement 45 filtres et 20 filtres. La taille des filtres utilisés dans toutes les couches convolutives citées précédemment est de $(4X1)$. Après cela, nous avons une couche de pooling moyenne (average-pooling) avec un pool size de 2. Ensuite, cette couche est connectée à trois couches convolutives 1D consécutives, qui comportent 45 filtres de taille de noyau $(4X1)$. La fonction ReLu est utilisée comme fonction d'activation pour toutes les couches convolutives précédentes. Ensuite, ces couches sont suivies de trois couches entièrement connectées (fully connected layers), qui incluent respectivement 128, 256 et 512 neurones, où la fonction d'activation ReLu est utilisée pour ces trois couches denses. Pour réduire l'effet de sur-ajustement (overfitting), nous avons ajouté deux couches Dropout (avec une probabilité marginale $p = 0,3$), après la première et la deuxième couche entièrement connectées. De plus, la deuxième régularisation normalisée (L2 norms) est utilisée pour toutes les couches entièrement connectées [41]. Enfin, la couche de sortie est une couche entièrement connectée, où le nombre de neurones est 5 (égal au nombre de classes d'environnements), d'où la fonction SoftMax est utilisée comme fonction d'activation. L'architecture proposée a été construite en utilisant la bibliothèque Tensorflow [1], où nous avons élaboré 20 époques avec une taille de lot de 50 (Batch size), pour entraîner le modèle.

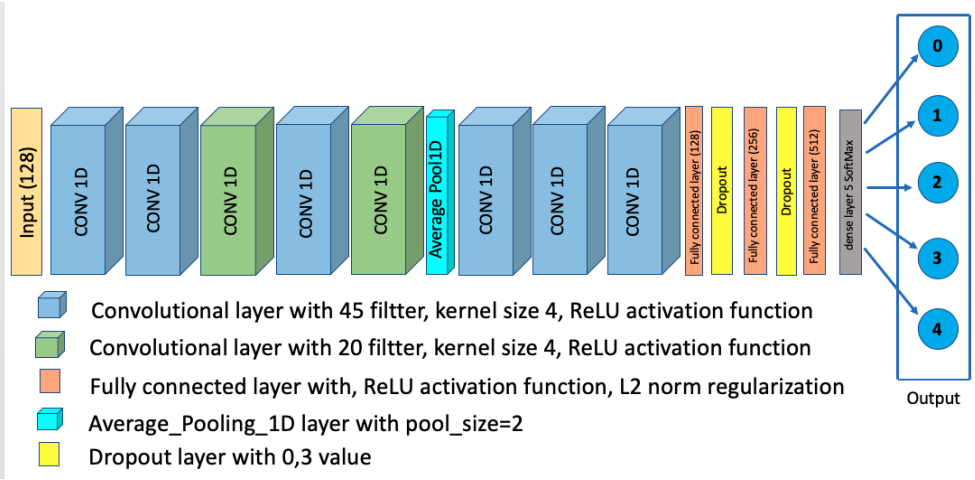


FIGURE 4.5: Architecture CNN proposée

4.4.1.2 Architecture ANN

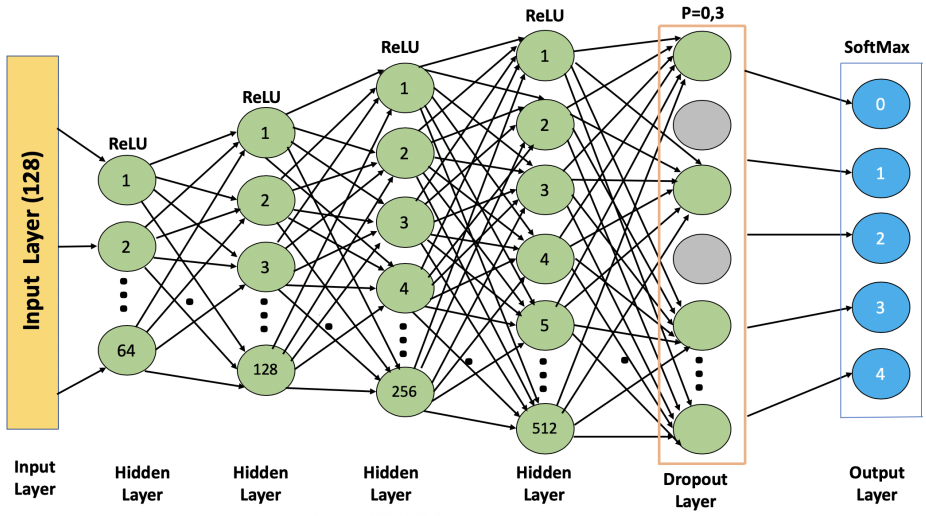


FIGURE 4.6: Architecture ANN proposée

L'architecture ANN élaboré est moins complexe que l'architecture CNN établie précédemment. Comme indiqué à la figure ci-après, elle est constituée de quatre couches denses entièrement connectées, qui comportent respectivement 64, 128, 256 et 512 neurones. La fonction ReLU est utilisée comme fonction d'activation pour les quatre couches denses. Ces couches entièrement connectées sont suivies d'une couche Dropout, configurée avec une probabilité marginale ($p = 0.3$). Enfin, on a la couche de sortie qui comporte 5 neurones (5 types d'environnements), qui utilise la fonction d'activation SoftMax.
Cette architecture a été construite en utilisant la bibliothèque Tensorflow. Pour entraîner le modèle, nous avons élaboré 20 époques avec une taille de lot de 50.

4.4.2 Génération de Data-Set

Étant donné que la tâche d'entraînement (Training) du modèle nécessite un ensemble de données (Data-Set). Dans cette dernière, nous considérons les 5 classes d'environnements véhiculaires : Rural LOS, Urban LOS, Urban NLOS, Highway LOS et Highway NLOS . Chaque environnement est modélisé par un canal sans fil basé sur des mesures réelles du retard, du gain et de la fréquence Doppler.
Les caractéristiques des canaux véhiculaires de chaque environnement peuvent être trouvées dans les Tableaux 3.1,3.3 et 3.2. Une étiquette (Label) est attribuée pour chaque environnement, qui sont les neurones de la dernière couche de sortie du modèle (Tableau 4.1).
Pour générer les échantillons de l'ensemble de données (Data-Set), nous effectuons une communication V2V half-duplex, basée sur la forme d'onde OFDM à l'aide de Matlab. Pour simuler les différents environnements véhiculaires (modèles de canaux sans fil) mentionnés ci-dessus, nous utilisons le framework V2VChannel de Matlab, qui est référencé à [3].
Nous transmettons plusieurs paquets 802.11p via les différents modèles de canaux. Pour chaque environnement, les paquets sont transmis, à des différentes valeurs du rapport signal/bruit (SNR), où la plage du SNR varie entre $15dB$ et $40dB$, avec un pas de $0,5dB$

Ce processus est répété 400 fois avec différentes versions du modèle de canal pour

Environnements Véhiculaires	**Label**
Highway NLOS	0
Highway LOS	1
Rural LOS	2
Urban LOS	3
Urban NLOS	4

TABLE 4.1: Labels des environnements véhiculaires

chaque environnement.
Dans chaque version, nous prélevons les LTS dans le paquet reçu et les valeurs CSI estimées. Ensuite, nous sauvegardons les 128 échantillons de données de LTS et CSI associés avec l'étiquette spécifique correspondant à chaque environnement comme indiqué dans le Tableau 4.1.
Les séquences enregistrées (F_i) pour les échantillons de données CSI ou LTS, peuvent être exprimées comme suit :

$$F_i = [[A(1), A(2), ..., A(N)]] \tag{4.2}$$

Où $A(i)$ représente l'échantillon CSI ou LTS dans chaque séquence.
À la fin du processus, nous avons collecté une Data-Set de taille de 100000 séquences LTS et 100000 séquences CSI dans les différents types d'environnement. 80% de cette Data-Set est utilisée comme ensemble d'apprentissage (Training-Set) et 20% est utilisée comme ensemble de validation (Validation-Set).

Notre Data-Set générée a été entraînée sur les deux architectures proposées, en utilisant la fonction de perte d'entropie croisée catégorique (categorical cross-entropy loss function) et l'algorithme d'optimisation Adam [40].
Le processus de training a été effectué sur une machine comprenant un GPU NVIDIA Tesla P100.

Tandis que le CSI et le LTS sont des données de forme complexes, nous considérons 3 configurations en fonction du format des données d'entrée de notre réseau de neurones. Nous avons utilisé les configurations suivantes :

— Le module des données complexes pour le premier banc d'essai.

— L'angle des données complexes pour le deuxième.

— Et une entrée à deux canaux pour le troisième, dans laquelle la partie réelle des nombres complexes est utilisée comme entrée du premier canal et la partie imaginaire est établie pour l'entrée du deuxième canal.

.

4.5 Évaluation des performances et résultats

Afin de vérifier la validité et l'exactitude des modèles proposés, nous évaluons leurs performances sur d'autres échantillons de tests (Test-Set), différents que celle utilisée dans le training (Data-Set). Ainsi, nous générons un ensemble d'échantillons de tests (Test-Set) en transmettant plusieurs paquets 802.11p à travers les différents modèles de canaux (framework V2VChannel de Matlab). Pour chaque environnement, la plage SNR est définie de $15dB$ à $40dB$, avec un pas de $0,25dB$.
Ce processus est répété 300 fois avec différentes versions du modèle de canal pour chaque environnement, afin d'obtenir des séquences de tests de 150000 LTS et 150000 CSI.

4.5.1 Évaluation des performances de l'approche basée sur LTS

Les résultats de taux précision de training et de test pour l'architecture CNN proposée, en utilisant les 3 configurations citées précédemment sont représentés au Tableau suivant :

	Précision de Training	Précision de Test
Module	96,02 %	92,22 %
Angle	95,92 %	91,78 %
2-Canaux	97,2 %	93,42 %

TABLE 4.2: Précision de training et de test de l'approche LTS pour les configurations d'entrées : Module, Angle et 2-Canaux.

Comme indiqué dans le Tableau 4.2, la configuration à 2-canaux obtient la meilleure précision de training par rapport aux deux autres configurations : module et angle. Où elle atteint $97,2\%$. De plus, en ce qui concerne la précision du test, nous pouvons également voir que la configuration à 2-canaux atteint $93,42\%$, ce qui est supérieur à la précision du test des configurations : module et d'angle, qui obtiennent respectivement $92,22\%$ et $91,78\%$.

La Figure 4.7 représente la matrice de confusion sur les échantillons de tests (Test-Set) pour l'architecture CNN proposée en utilisant les séquences LTS comme entrées du modèle CNN . Nous avons utilisé une configuration d'entrée à deux canaux, tant dite qu'elle donne de meilleures performances de précision comparant aux autres configurations.
À partir de cette matrice de confusion, nous pouvons voir que notre modèle CNN proposé est capable de reconnaître de manière fiable les différents environnements véhiculaires, où il identifie correctement les environnements : H-NLOS, H-LOS avec une précision individuelle par classes de $98,3\%$, $86,7\%$ respectivement. Et aussi, il identifie également les environnements R-LOS, U-LOS et U-NLOS avec des précisions individuelles par classes autour de 94%.
Nous avons utilisé également les échantillons de tests (Test-Set) pour évaluer l'architecture ANN décrit à la section 4.4.1.2. Les résultats sont représentés à la matrice de confusion dans la Figure 4.8. D'après cette dernière on voit clairement qu'en terme de précision de test global l'architecture CNN fournit un taux de précision de test de l'ordre 93.42%, qui est plus significatif que celui de l'architecture ANN qui obtient un taux de précision de test de 86.19%.

Accuracy 93.42%

	H-NLOS	H-LOS	R-LOS	U-LOS	U-NLOS
H-NLOS	98.3%	3.2%	0%	0.8%	5.7%
H-LOS	0.4%	86.7%	0.3%	2.7%	0.2%
R-LOS	0%	2.6%	94.1%	0.3%	0%
U-LOS	0%	5.8%	4.7%	94%	0.1%
U-NLOS	1.3%	1.7%	0.9%	2.2%	94%

FIGURE 4.7: Matrice de confusion basée sur les séquences LTS pour l'architecture CNN proposée sur les échantillons de tests (Test-Set)

Aussi on peut constater qu'en termes des précisions individuelles pour chaque environnement, l'architecture CNN obtient des taux précisions plus élevés pour tous les types d'environnements véhiculaires (H-NLOS, H-LOS, R-LOS, U-LOS et U-NLOS).

Accuracy 86.16%

	H-NLOS	H-LOS	R-LOS	U-LOS	U-NLOS
H-NLOS	90,1 %	1%	0.6%	1.6%	5.4%
H-LOS	1.8%	88.3%	12.6%	4.1%	1.6%
R-LOS	0 %	6.2 %	74.7%	3.2%	0.3%
U-LOS	1.9 %	2.4%	11.5%	88.1%	3.1%
U-NLOS	6.2%	2.1%	0.6%	3%	89.6%

FIGURE 4.8: Matrice de confusion basée sur les séquences LTS pour l'architecture ANN sur les échantillons de tests (Test-Set)

En outre, notre modèle CNN proposé a été comparé à d'autres classificateurs candidats d'apprentissage automatique, qui sont : le classificateur Random Forest (RF, avec 100 arbres), le classificateur K-Neighbours (K-NN), où K a été défini sur 5 voisins, le classificateur Gaussian Naive Bayes (GNB) et un classificateur basé sur Support Vector Machine (SVM), avec un noyau linéaire.
Le Tableau 4.3 montre une comparaison entre notre modèle CNN proposé et les autres approches mentionnées précédemment. nous prenons en compte la précision du test globale et aussi le temps de prédiction moyen de l'environnement. Ce temps de prédiction a été calculé en utilisant un GPU NVIDIA Tesla P100.
En ce qui concerne la table 4.3, il est évident qu'en terme de temps moyen de prédiction, notre architecture CNN proposée obtient de meilleures performances que SVM et K-NN, où elle fournit un temps de prédiction de $51,33\mu s$. Ce temps est comparable aux autres approches (ANN, RF, GNB), qui ont un temps de prédiction moins significatif, cependant, la précision de test globale de notre modèle CNN est largement plus grande que ces approches, qui est égale à $93,42\%$.

Approche	Précision (%)	Temps de prédiction (μs)
CNN proposée	93.42	51.33
ANN	86.16	23.11
RF	68.34	25.71
K-NN	63.18	7180
GNB	20.62	4.11
SVM	31.38	10499

Table 4.3: Comparaison de Précision de classification et de temps moyens de prédiction pour l'approche LTS

Afin de fournir plus de détails sur la précision de test de classification des environnements, les matrices de confusion des approches considérées sont présentées dans les Figures de 4.7 à 4.12.

D'après ces matrices de confusion, nous pouvons voir que les approches K-NN et RF peuvent identifier les environnements H-NLOS et U-LOS, avec une précision de test individuelle acceptable (jusqu'à 80%). Ces deux approches fournissent moins de 65% de précision de test individuel pour les environnements : H-LOS, R-LOS et U-LOS.

D'après les figures 4.11 et 4.12, il est clair que les deux classificateurs GNB et SVM ne parviennent pas à donner une identification d'environnement fiable.

Accuracy 63.18%

	H-NLOS	H-LOS	R-LOS	U-LOS	U-NLOS
H-NLOS	88.9%	5.8%	3.6%	4.6%	12.1%
H-LOS	0.3%	44.9%	26.5%	16.5%	0.9%
R-LOS	0.7%	21.1%	44.7%	11.8%	0.5%
U-LOS	2.1%	19.5%	21.8%	55.5%	4.6%
U-NLOS	8%	8.7%	3.4%	11.5%	81.9%

FIGURE 4.9: Matrice de confusion basée sur les séquences LTS pour l'approche KNN sur les échantillons de tests (Test-Set)

Accuracy 68.34%

	H-NLOS	H-LOS	R-LOS	U-LOS	U-NLOS
H-NLOS	89.7%	1.6%	1.3%	2.5%	7.1%
H-LOS	1.6%	49.8%	25.8%	20%	1.8%
R-LOS	0.3%	22.1%	53.2%	9.6%	0.1%
U-LOS	3.4%	22%	16.8%	63.7%	5.7%
U-NLOS	5%	4.5%	2.9%	4.1%	85.3%

FIGURE 4.10: Matrice de confusion basée sur les séquences LTS pour l'approche RF sur les échantillons de tests (Test-Set)

Accuracy 20.62%

	H-NLOS	H-LOS	R-LOS	U-LOS	U-NLOS
H-NLOS	25.9%	29.7%	5.3%	13.9%	23.8%
H-LOS	15.6%	5.2%	30.6%	20.9%	21.1%
R-LOS	16.6%	1.1%	31.1%	24.6%	14.3%
U-LOS	17.8%	35.7%	25.4%	19.2%	19.1%
U-NLOS	24%	28.3%	7.6%	21.4%	21.7%

FIGURE 4.11: Matrice de confusion basée sur les séquences LTS pour l'approche GNB sur les échantillons de tests (Test-Set)

Accuracy 31.38%

	H-NLOS	H-LOS	R-LOS	U-LOS	U-NLOS
H-NLOS	24%	13.3%	4.6%	42.5%	24.8%
H-LOS	16.1%	30.6%	32%	8.2%	10.9%
R-LOS	17.3%	18.5%	38%	4.9%	11.4%
U-LOS	20.3%	24.5%	16.4%	29.4%	15.5%
U-NLOS	22.3%	13.2%	9.1%	15%	37.4%

FIGURE 4.12: Matrice de confusion basée sur les séquences LTS pour l'approche SVM sur les échantillons de tests (Test-Set)

4.5.2 Évaluation des performances de l'approche basée sur CSI

Comme indiqué précédemment l'identification de l'environnement véhiculaire, en se basant sur les CSI estimées comme entrée pour le modelé CNN proposé a été évaluée, en utilisant 3 configurations : deux canaux, le module et l'angle.
Les résultats de précision de training et de test sont présentés au Tableau 4.4. D'après ce tableau comparatif, il est clair que la configuration des caractéristiques des entrées à deux canaux fournit les meilleures performances, où elle obtient une précision de test de $96,48\%$. De plus, cette précision de test obtenue est supérieure à la précision fournie par l'approche LTS qui est égale à $93,42\%$ en configuration à deux canaux.

	Précision de Training	Test Accuracy
Module	96,65 %	90,63 %
Angle	96,54 %	91,50 %
2-canaux	99,10 %	96,48 %

TABLE 4.4: Précision de training et de test de l'approche CSI pour les configurations d'entrées : Module, Angle et 2-Canaux.

La figure 4.13, représente la matrice de confusion pour l'architecture CNN proposée (Figure 4.5) sur l'ensemble de Test-Set, dans le cadre de l'utilisation des valeurs CSI comme caractéristiques d'entrée pour le modèle. Les résultats présentés ont été calculés en tenant compte d'une forme d'entrée à deux canaux car elle fournit les performances les plus précises.

Accuracy 96.48%

	H-NLOS	H-LOS	R-LOS	U-LOS	U-NLOS
H-NLOS	99.9%	0.3%	0%	0%	2.6%
H-LOS	0.1%	95.2%	2.9%	1.8%	0%
R-LOS	0%	0%	92.7	0.2%	0%
U-LOS	0%	4.2%	4.1%	97.4%	0.2%
U-NLOS	0%	0.3%	0.3%	0.6%	97.2%

FIGURE 4.13: Matrice de confusion basée sur l'approche CSI pour le modèle CNN proposé

De cette matrice de confusion, il est clairement visible que notre modèle proposé basé sur les valeurs CSI peut identifier de manière fiable tous les environnements véhiculaires, avec une précision individuelle par classe supérieure à 92%. Où il atteint $99,9\%$, $95,2\%$, $92,7\%$,$97,4\%$ et $97,2\%$ pour les environnements : H-NLOS, H-LOS, R-LOS, U-LOS et U-NLOS respectivement.

Le modèle CNN proposé basé sur CSI comme caractéristiques d'entrées était comparé à l'apprentissage automatique des classificateurs concurrents : RF, K-NN, GBN, SVM, et également comparé à l'architecture ANN (sous-section 4.4.1.2), en termes de précision de tests et de temps moyen nécessaire pour identifier l'environnement (temps de prédiction, obtenue en utilisant un GPU NVIDIA Tesla P100).
L'architecture ANN et les paramètres définis de ces classificateurs sont les mêmes que ceux décrits précédemment dans la sous-section 4.5.1.

Approche	Précision(%)	Temps Moyen de Prédiction (μs)
CNN proposée	96.48	39.56
ANN	85.64	21.11
RF	67.77	24.04
K-NN	59.26	8999
GNB	27.06	4.38
SVM	32.33	15756

Table 4.5: Comparaison de Précision de classification et de temps moyens de prédiction pour l'approche CSI

D'après le Tableau 4.3, le modèle CNN proposé obtient de meilleures performances que SVM et K-NN, en ce qui concerne le temps de prédiction, où il revient à $39,56\mu s$. Ce temps de prédiction atteint est comparable aux autres approches (ANN, RF, GNB), qui ont un temps de prédiction faible. Cependant, en terme de précision du test, notre modèle CNN surpasse largement toutes les autres approches, où il atteint $96,48\%$.

Pour plus de détails sur la précision de test individuelle par classe, les matrices de confusion des approches RF, K-NN, GBN, SVM et ANN sont présentées dans les Figures de 4.14 à 4.18

Accuracy 85.64%

	H-NLOS	H-LOS	R-LOS	U-LOS	U-NLOS
H-NLOS	94.5%	2%	0.6%	1.7%	3.5%
H-LOS	1.6%	82.4%	15.2%	5.1%	1.1%
R-LOS	0.1%	9.9%	76.6%	6.2%	0%
U-LOS	1%	4.1%	7.1%	82.6%	2.1%
U-NLOS	2.8%	1.7%	0.5%	4.4%	93.3%

FIGURE 4.14: Matrice de confusion basée sur les valeurs CSI pour le modèle ANN

Accuracy 67.77%

	H-NLOS	H-LOS	R-LOS	U-LOS	U-NLOS
H-NLOS	86.9%	1.8%	1.9%	2.8%	5.8%
H-LOS	2.1%	50.4%	28.2%	17%	0.8%
R-LOS	0.9%	24%	54.2%	11.7%	0%
U-LOS	5.7%	20/2%	13.1%	62.2%	6.3%
U-NLOS	4.5%	3.6%	2.6%	6.2%	87.1%

FIGURE 4.15: Matrice de confusion basée sur les valeurs CSI pour l'approche RF

Accuracy 59.26%

	H-NLOS	H-LOS	R-LOS	U-LOS	U-NLOS
H-NLOS	91.1%	5.3%	5%	7.4%	9.9%
H-LOS	0.1%	41.3%	29%	14.5%	0.4%
R-LOS	0.6%	23.5%	43.5%	10.7%	0.2%
U-LOS	1%	23%	18.7%	54.5%	4.6%
U-NLOS	7.2%	6.9%	3.8%	12.9%	84.9%

FIGURE 4.16: Matrice de confusion basée sur les valeurs CSI pour l'approche K-NN

Accuracy 27.06%

	H-NLOS	H-LOS	R-LOS	U-LOS	U-NLOS
H-NLOS	26.5%	17.1%	3.7%	26%	19.7%
H-LOS	17.1%	19%	30%	12.4%	19%
R-LOS	16.6%	22.8%	33.3%	0%	13.3%
U-LOS	18%	22.6%	23.4%	31.6%	19.8%
U-NLOS	21.8%	18.6%	9.6%	30%	28.3%

FIGURE 4.17: Matrice de confusion basée sur les valeurs CSI pour l'approche GNB

Accuracy 32.33%

	H-NLOS	H-LOS	R-LOS	U-LOS	U-NLOS
H-NLOS	29.3%	7%	2.1%	18.9%	22%
H-LOS	15.9%	31.8%	29.3%	19.7%	14.6%
R-LOS	15.7%	26%	44.7%	13.7%	11.8%
U-LOS	18.6%	25.5	19.4%	24.7%	17.3%
U-NLOS	20.5%	9.8%	4.4%	23.1%	34.3%

FIGURE 4.18: Matrice de confusion basée sur les valeurs CSI pour l'approche SVM

Selon les résultats d'évaluation des performances représentées dans la sous-section 4.5.2 et la sous-section 4.5.1, on peut conclure que l'utilisation des valeurs CSI comme caractéristiques d'entrée pour le modèle CNN proposé est plus précise que l'utilisation des échantillons LTS, en termes de précision de test et de temps moyen de prédiction de l'environnement.

4.5.3 Comparaison des performances de notre modèle CNN avec des architectures existantes

Dans le cadre de l'évaluation des performances, notre architecture CNN proposée a été comparée aux architectures de classification de pointes populaires suivantes : ResNet50 [29], Xception [16], InceptionV3 [74], InceptionResNetV2 [73], DenseNet201 [32] and MobileNetV2 [60].
Pour cela Nous avons entraîné notre Data-Set constituer des séquences CIS sur ces architectures, puis nous avons évalué leurs performances de classification sur les séquences CSI de base de données de test (Test-Set).
Avant ce processus, nous avons modifié la forme d'entrée de la première couche d'entrée pour l'adapter à nos entrées de données et nous avons mis également à jour la taille de la couche de sortie à 5 classes, ce qui est égal au nombre de type de l'environnement véhiculaire à identifier.
Étant donné que les architectures mentionnées précédemment ont été essentiellement conçues pour prendre 2 dimensions (2D) comme taille de forme d'entrée, nous avons considéré une matrice de canal 2D au lieu d'un vecteur de canal 1D. Ainsi, nous réorganisons nos séquences CSI de notre Data-Set et notre Test-Set de 1D à 2D comme suit :

$$\underset{[n*n]}{H_{2D}} = Diag(\underset{[1*n]}{H_{1D}}) \tag{4.3}$$

Où $Diag()$ est la matrice diagonale, H_{2D} et H_{1D} sont respectivement la matrice de canal et le vecteur de canal correspondant, où leurs coefficients sont les valeurs CSI. n

est égal à 128, ce qui équivaut au nombre de valeurs CSI estimées par paquet. Ainsi, nous avons une entrée de taille 128×128.
Dans le tableau 4.6, nous avons comparé notre modèle CNN proposé aux architectures de pointe indiquées, en termes de précision de test moyenne (Acc), la précision de test individuelle pour chaque environnement et le temps moyen nécessaire pour identifier l'environnement (temps de prédiction), où ce temps de prédiction a été calculé en utilisant un GPU NVIDIA Tesla P100.

D'après le tableau 4.6, il est clair que notre modèle proposé obtient les meilleures performances en terme de temps moyen de prédiction, par rapport à toutes les autres architectures présentées dans le tableau, où il obtient un temps de prédiction de $39,56\mu s$. Concernant les architectures : ResNet50, Xception et InceptionV3 ils avaient un temps de prédiction autour de $700\mu s$, où les architectures InceptionResNetV2, DenseNet201, MobileNetV2 et DCNN [21] obtiennent : $1621\mu s$, $1349\mu s$, $318\mu s$ et $125\mu s$, respectivement. Ainsi le temps de prédiction atteint par notre modèle est significativement inférieur au temps de prédiction atteint par les autres architectures (au moins 3 fois moins).

En ce qui concerne la précision globale du test, notre modèle atteint $96,48\%$. Cette précision obtenue est plus grande que la précision de test obtenue par les architectures : ResNet50, Xception, InceptionV3, InceptionResNetV2, DenseNet201 et MobileNetV2, qui obtiennent $89,54\%$, $91,32\%$, $91,08\%$, $89,98\%$, $92,94\%$ et $78,86\%$, respectivement. Bien que l'architecture DCNN proposée dans [21] obtienne un peu plus que notre modèle en termes de précision de test ($0,54\%$ de plus), le temps moyen de prédiction atteint par notre modèle est de l'ordre de 3 fois inférieur à celui de DCNN [21].

Pour donner plus de détails sur les performances de classification des architectures considérées, la précision de test individuelle de chaque environnement véhiculaire est présentée dans le tableau 4.6. Nous pouvons voir que toutes les architectures peuvent discriminer avec succès l'environnement H-NLOS, avec une précision de plus de 96%, où notre modèle obtient la meilleure précision, qui est égale à $99,9\%$.
Concernant l'environnement H-LOS, les précisions individuelles sont acceptables (proches de 80%), pour MobileNetV2 et InceptionV3, où les autres architectures fournissent de bonnes précisions individuelles (environ 90%).

Les précisions de tests pour l'environnement R-LOS sont moins significatives par rapport aux autres environnements, cela est dû aux caractéristiques de canal dans cet environnement qui sont proches des caractéristiques de canal des environnements U-LOS et H-LOS. Cependant, les précisions pour l'environnement R-LOS sont toujours considérées comme de bonnes performances pour notre modèle, MobileNetV2 et DCNN qui sont autour de 90%. Pour les autres modèles La précision individuelle pour cet environnement est acceptable (environ 80%).
Pour l'environnement U-LOS, les précisions de tests sont autour de 90%, où notre modèle proposé obtient la meilleure précision, ce qui est égal à $97,4\%$.
Concernant l'environnement U-NLOS, à part l'architecture MobileNetV2, toutes les autres architectures peuvent reconnaître cet environnement avec une grande précision (jusqu'à 93%).

Architecture	H-NLOS Acc (%)	H-LOS Acc (%)	R-LOS Acc (%)	U-LOS Acc (%)	U-NLOS Acc (%)	Acc(%)	Temps Moyen de Prédiction (μs)
Notre Modéle CNN	99.9	95.2	92.7	97.4	97.2	96.48	39.56
ResNet50	98.1	88.2	77.8	90.1	93.5	89.54	672
Xception	97.8	91.7	81.4	91.2	94.5	91.32	794
InceptionV3	99.1	79.8	86.9	96.1	93.9	91.08	683
Inception ResNetV2	98.5	89.1	80	86.5	95.8	89.98	1621
DenseNet201	98.5	92.7	85.7	91.2	96.6	92.94	1349
MobileNetV2	96.8	77.8	96	58.2	65.5	78.86	318
DCNN [21]	98.9	96.9	94.3	95.8	99.2	97.02	125

TABLE 4.6: Comparaison de notre modèle CNN aux architectures de pointes

4.5.4 Fiabilité de performance minimaux

Étant donné que notre approche d'identification de l'environnement véhiculaire proposée basée sur les valeurs CSI estimées, est destinée à être mise en œuvre dans des voitures connectées autonomes, pour une utilisation dans un contexte critique, il est nécessaire que le temps moyen de prédiction du type de l'environnement réponde systématiquement aux exigences de latence dans chaque scénario. Selon [19], cette limite de temps critique est généralement définie en quelques millisecondes. Notre modèle d'identification d'environnement basé sur les CSI a démontré un temps moyen de prédiction de manière cohérente en **microsecondes**, beaucoup moins du temps requis comme on le voit dans les résultats présentés.
En plus de cela, notre architecture CNN proposée utilisant les valeurs CSI comme caractéristiques d'entrée a prouvé qu'elle pouvait identifier de manière fiable les différents environnements de véhicules avec une grande précision de test globale $(96,48\%)$ qui répondent aux exigences des voitures connectées autonomes.

4.6 Conclusion

Dans ce chapitre, Nous avons présenté une nouvelle approche d'identification de l'environnement véhiculaire basée sur le Deep Learning, en utilisant le canal sans fil véhiculaire sous la forme des CSI estimées, comme caractéristique d'entrée pour notre modèle CNN. Les résultats de notre validation de test ont démontré que notre méthode peut reconnaître de manière fiable l'environnement véhiculaire dans lequel le véhicule circule, avec une grande précision $(96,48\%)$. Ces résultats ont également montré que notre approche a un temps de prédiction mesurée en microsecondes, bien dans la plage opérationnelle attendue dans les divers scénarios de conduite autonome. De plus, notre modèle CNN a des performances comparables aux architectures de pointes.

Chapitre 5

Sécurité des communications véhiculaires

Sommaire

5.1 Introduction 84
5.2 État de l'art sur la sécurité pour les communications véhiculaires. 85
5.2.1 Solutions basées sur les clés partagées 85
5.2.2 Solutions basées sur les clés générées à base de la couche physique 85
5.2.3 Solutions basées sur l'informatique quantique 86
5.3 Processus de génération de clés 87
5.4 Algorithme proposé 87
5.4.1 Algorithme de génération de clés basé sur RSSI 88
5.4.2 Algorithme de génération de clés proposé basé sur CSI . . 89
5.5 Tests expérimentaux 90
5.5.1 Collecte des données et pré-traitement 91
5.5.1.1 Méthode basée sur les RSSI 91
5.5.1.2 Méthode basée sur les CSI 92
5.5.2 Résultats et pourcentage de clés similaires pour les RSSI . 93
5.5.3 Résultats et pourcentage de clés similaires pour les CSI . 95
5.6 Évaluation des performances de l'approche CSI 98
5.6.1 Min-entropie 98
5.6.2 Temps de génération de clés 98
5.6.3 complexité de l'algorithme 99
5.6.4 Attaque non-intrusive 100
5.6.5 Attaque de brute force 101
5.6.6 Classification du générateur de bits aléatoires suivant les standards de NIST 102
5.6.7 Comparaison avec l'état de l'art 103
5.7 Conclusion 104

5.1 Introduction

Étant donné le rôle essentiel que jouent les communications véhiculaires (V2X) dans la gestion de trafic et ainsi dans la protection des usages de la route, la sécurité de ces communications est quasi importante. Les véhicules ITS sont très mobiles et échangent des informations de manière continue, cependant, la connectivité croissante entre les véhicules (V2V) et entre les véhicules et l'infrastructure (V2I) engendre une augmentation des surfaces d'attaque et des vulnérabilités [66]. En conséquence, diverses études [2], [28], [4], [25], [14], ont confirmé les préoccupations de la sécurité des ITS suivant les standards exigés comme suit :

— **L'authentification :** qui garantit que les messages soient générés par des utilisateurs légitimes.

— **La disponibilité :** qui maintient l'accès aux services ITS.

— **L'intégrité :** qui empêche la falsification des messages.

— Et **la confidentialité :** qui limite l'accessibilité aux messages par les pirates.

Actuellement, les communications véhiculaires sont toujours vulnérables aux attaques, de ce fait la sécurité des communications V2X est un défi de recherche ouvert et nécessaire à l'avancement des travaux de déploiement des ITSs.
Dans le cadre de nos travaux de recherche dans ce domaine nous proposons dans ce chapitre un algorithme de génération de clés de sécurité pour les communications dans les systèmes de transports intelligents, en se basant sur les caractéristiques du canal de transmission. Ces travaux de recherche ont été élaborés en collaboration avec l'université d'Irvine en Californie (UCI).

5.2 État de l'art sur la sécurité pour les communications véhiculaires.

Suivant la littérature les solutions de sécurité dédiées aux communications véhiculaires peuvent se diviser en trois types

5.2.1 Solutions basées sur les clés partagées

Actuellement, les méthodes qui se basent sur des clés partagées pour sécuriser les communications V2X utilisent les PKI (Public key infrastructurs) [37] .Cependant, ces méthodes de cryptographie basées sur les PKI traditionnelle présentent certains risques [20] et encourent une latence importante lors de l'exécution des opérations cryptographiques nécessaires de l'ITS-S [27]. De plus, elles nécessitent une quantité importante de ressources et d'énergie [57]. Pour les applications V2X où les communications critiques pour la sécurité doivent être terminées dans les 200 ms et où du matériel embarqué avec une puissance de calcul limitée est utilisé, une telle latence élevée et des besoins en énergie ne sont pas viables [9]. Ces restrictions ont conduit au développement d'une infrastructure à clés publiques véhiculaires (VPKI).
Le SCMS (Security Credential Management System) est un candidat pour la standardisation de la sécurité V2X aux États-Unis car il peut fournir une authentification des données pour préserver la confidentialité [69] [12]. Malgré ces mesures de sécurité, l'évaluation des risques a conclu que le SCMS est toujours vulnérable à certains types d'attaques [12].

5.2.2 Solutions basées sur les clés générées à base de la couche physique

La génération des clés à base de la couche physique consiste à utiliser les caractéristiques du canal de communication entre l'émetteur et le récepteur, comme source d'entropie pour générer des bits aléatoires. Dans cette approche basée sur la couche physique, des mesures du canal sans fil partagé sont effectuées séparément dans chaque partie communicante (véhicule 1 et véhicule 2,) pour amorcer un générateur de clé individuelle. Le canal partagé signifierait que des clés symétriques équivalentes sont générées sans nécessiter une étape d'échange de clés (Voir figure 5.1). Les travaux de recherches existants ont exploré les différents attributs des canaux comme sources d'entropie aléatoire. La majorité de ces méthodes sont basées sur des mesures de RSSI (Received Signal Strength Indicator), ou sur la réponse impulsionnelle de canal CIR (Channel Impulse Respons), qui peut être caractérisée par les états du canal CSI estimés. Une technique de génération de clés basées sur les mesures RSSI pour la communication V2V est introduite dans [81]. Dans cette méthode, les valeurs RSSI sont mesurées à partir d'un signal de sondage échangé entre deux véhicules communicants via Bluetooth. Les valeurs RSSI sont ensuite quantifiées en utilisant un seuil prédéfini pour obtenir une séquence de valeurs binaires, qui sert à générer des clés pour sécuriser les communications. Une autre approche similaire a été proposée dans [85] pour sécuriser les communications optiques entre les véhicules. Cette approche a été étendue à l'application de la sécurité de liaison par fibre optique dans [86] et [87], où la dispersion du mode de polarisation (PMD) a été utilisée comme source aléatoire pour générer des clés à partir des signaux optiques. Dans cette méthode, la PMD a été divisée en plusieurs tailles de groupes, puis quantifiée en utilisant des seuils supérieurs et inférieurs pour chaque taille de groupe pour obtenir des séquences binaires utilisées pour la génération de clés.
Dans [84], une nouvelle méthode d'extraction des bits en utilisant les CSI dans un système 4G-LTE est présentée. Elle vise à estimer les CSI de liaison montante de la station de base (BS) et de l'équipement utilisateur (UE) en même temps en échangeant un signal de sondage privé connu dans les deux parties communicantes.

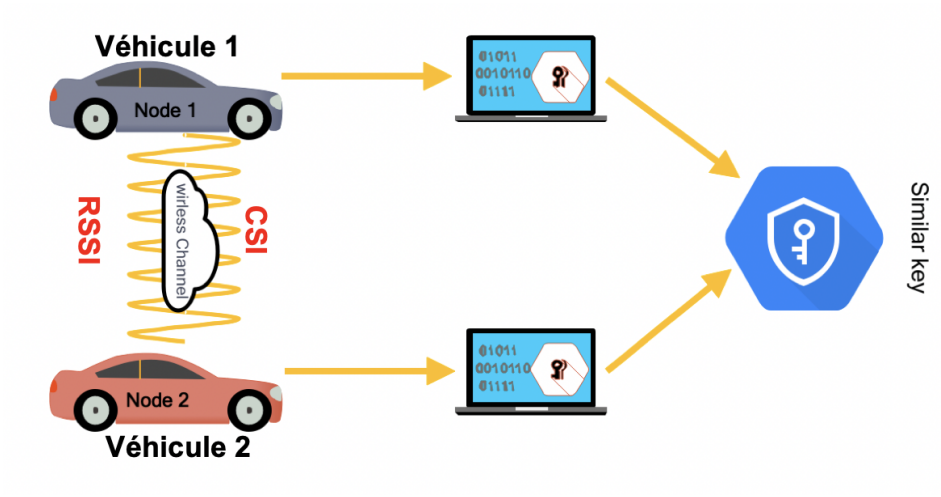


FIGURE 5.1: Principe de génération des clés en utilisant les caractéristiques du canal.

L'amplitude des CSI estimés est ensuite quantifiée via un seuil pour obtenir une succession de bits, qui seront utilisés pour générer des clés.
Une approche appelée SKECE a été explorée dans [90]. Dans cette méthode l'extraction de clé comporte trois étapes principales. Premièrement, les valeurs CSI sont quantifiées en fixant deux seuils adaptatifs et en attribuant «1» aux valeurs au-dessus du seuil supérieur et «0» aux valeurs inférieures au seuil inférieur. Les valeurs CSI entre les seuils sont supprimées. Deuxièmement, les flux binaires issus de la quantification sont hachés pour renforcer la sécurité. Enfin, une réconciliation est effectuée pour extraire un train binaire cohérent en utilisant une méthode de recombinaison de clé pondérée.
Un nouveau mécanisme d'extraction de clé appelé quantification de canal alternatif a été établi dans [79] et [80]. Il repose sur l'utilisation de secteurs de quantification non équivalents sur l'espace des canaux complexes observés. Semblable à la quantification du canal alterné, un mécanisme intelligent de génération de clés appelé déphasage est exploré dans [67]. Il s'agit de convertir les phases décalées des valeurs de CSI en points de constellation.
Par rapport aux approches basées sur les mesures RSSI, les méthodes basées sur les CSI ont démontré des avantages importants au niveau de la confidentialité et de taux de génération de clés [44]. Cependant, l'approche basée sur les RSSI reste toujours une méthode à accès facile, car elle est accessible à partir de couches réseau supérieures et est généralement auto-implémentée dans les plates-formes SDR.

5.2.3 Solutions basées sur l'informatique quantique

L'informatique quantique opte à faire une révolution dans le domaine de la sécurité des systèmes de communication. De ce fait les solutions de pointe actuelles dans ce domaine de recherche sont basées sur la cryptographie quantique, qui consiste à utiliser la théorie quantique pour effectuer des tâches cryptographiques. Beaucoup de recherches ont été menées dans ce domaine comme : les sources d'entropie basées sur la génération des nombres aléatoires quantiques [47], et le système de gestion des informations d'identification de sécurité quantique (quantum-SCMS) [7]. Cependant, ces approches quantiques sont des solutions à coût très élevé.

5.3 Processus de génération de clés

La plupart des méthodes de génération de clés existantes utilisent les RSSI ou les CSI comme source d'entropie, suivent généralement le processus illustré par la figure 5.2. Ce processus comporte les trois étapes suivantes :

1. **Quantification :** Il s'agit du processus de conversion des valeurs RSSI mesurée ou les CSI estimées en une séquence de bits. Comme indiqué précédemment, dans la littérature différents quantificateurs ont été proposés en se basant sur des seuils supérieurs et inférieurs fixes ou dynamiques des valeurs moyennes des RSSI mesurées ou les CSI estimées.

2. **Réconciliation :** Cette étape vise à améliorer la corrélation croisée des caractéristiques du canal utilisé (RSSI ou CSI), entre deux nœuds communicants en corrigeant les écarts dans les clés dérivées. De nombreux protocoles de réconciliation ont été utilisés dans les travaux de recherche, les méthodes les plus précises sont détaillées dans [68].

3. **Amplification de la confidentialité :** Lors de l'étape de réconciliation, certaines informations non cryptées sont transmises, ce qui peut potentiellement divulguer des informations sur la clé secrète. Pour résoudre ce problème, l'amplification de la confidentialité est utilisée pour augmenter la propriété aléatoire de la clé, en atténuant les effets de toute fuite potentielle. Cela peut être fait avec un extracteur [48] ou avec l'utilisation des fonctions universelles de hachage [57] [35].

FIGURE 5.2: Processus de génération de clés.

5.4 Algorithme proposé

Dans le cadre de nos travaux de recherche, nous avons élaboré deux algorithmes de génération de clés pour sécuriser les communications véhiculaires comme suit :

— **Algorithme 3 :** Se base sur les mesures RSSI, avec une quantification classique, qui utilise des seuils inférieurs et supérieurs pour obtenir une séquence binaire.

— **Algorithme 4 :** Se base sur les CSI estimées, avec une nouvelle méthode de quantification en utilisant un démodulateur QAM pour convertir les valeurs CSI en suite binaire.

5.4.1 Algorithme de génération de clés basé sur RSSI

Algorithm 3: Algorithme de génération de clés basé sur RSSI

Input: Measured Signal Strength RSS
Input: Sample Time Step : τ_{step}
Input: Group Size : G_{size}
Input: Threshold parameter : α
Input: Required Key Length : L_{key}
Output: Generated Key : Key

1 $L = 0; Key = 0; RSS_{set} = \emptyset; RSS_{filtered} = \emptyset; Key_{idx} = \emptyset$
2 **while** L ¡ L_{key} **do**
3 **for** $i = 1$ **to** G_{size} **do**
4 $RSS_{set} = RSS_{set} \bigcup RSS$;
5 $Wait(\tau_{step})$
6 $RSS_{filtered} = RSS_{set} * H_{highpass}(t)$;
7 $MeanValue = AverageValueofRSS_{filtered}$;
8 $Var = VariationValueofRSS_{filtered}$;
9 $Th_{up} = MeanValue + \alpha * Var$;
10 $Th_{lo} = MeanValue - \alpha * Var$;
11 **foreach** $RSS_j \in RSS_{filtered}$ **do**
12 **if** $RSS_j > Th_{up}$ **then**
13 $Key = (Key << 1) + 1$;
14 $L = L + 1$;
15 Record j in Key_{idx};
16 **else if** $RSS_j < Th_{lo}$ **then**
17 $Key = (Key << 1) + 0$;
18 $L = L + 1$;
19 Record j in Key_{idx};
20 Exchange Key_{idx}; Remove mismatch bits from Key;
21 Return Key;

L'algorithme 3 a été inspiré de [82], son processus comporte les trois étapes indiquées à la figure 5.2 : quantification, réconciliation et amplification de la confidentialité. Son principe de fonctionnement est le suivant :

— Commence par prendre G_{size} et τ_{step} pour collecter les valeurs RSSI du canal de transmission comme indiqué aux lignes de 3 à 5 dans Algorithme 3. G_{size} est le nombre de groupes de valeurs RSSI à quantifier à chaque fois et τ_{step} indique le temps entre deux mesures consécutives de RSSI.

— Après avoir collecté les mesures RSSI, on applique un filtre passe-haut pour éliminer le bruit basse fréquences (ligne 6 dans Algorithme 3).

— Ensuite on calcule les seuils inférieurs Th_{lo} et supérieurs Th_{up} de quantification, pour les valeurs $RSS_{filtered}$ comme indiqué aux lignes de 7 à 10 dans 3.

— Par la suite, les lignes de 11 à 19 consistent à générer la clé Key après avoir quantifié les RSSI tout en enregistrant également les indices des valeurs RSSI utilisées. Cela sera utilisé plus tard dans l'étape de réconciliation (ligne 20) pour supprimer les bits associés à des indices incompatibles.

5.4.2 Algorithme de génération de clés proposé basé sur CSI

Le processus de notre méthode proposé de génération de clés basé sur CSI (voir Algorithme 4) comprend seulement deux étapes principales : la quantification et l'amplification de la confidentialité. Dans l'approche que nous proposons, l'étape de réconciliation typique n'est pas nécessaire.

Algorithm 4: Algorithme de génération de clés basée sur CSI

Input: Channel State Information CSI
Input: Required Key Length : L_{key}
Output: Generated Key : Key

1 $L = 0; Key = 0; CSI_{set} = \emptyset; Verified = False$
2 **while** $not\ Verified$ **do**
3 $\quad GeneratedKeys_{set} = \emptyset$
4 $\quad CSI_{set} = QAMDemodQuantizer(CSI)$
5 $\quad$ **for** $i = 1$ **to** $Length(CSI_{set})$ $in\ StepSize = L_{key}$ **do**
6 $\quad\quad GeneratedKeys_{set} = GeneratedKeys_{set} \bigcup CSI_{set}[i : i + L_{key}]$
7 $\quad$ **foreach** $TestKey \in GeneratedKeys_{set}$ **do**
8 $\quad\quad$ Exchange verification packets ;
9 $\quad\quad$ **if** *Verification Exchange Returns True* **then**
10 $\quad\quad\quad Verified = True$
11 $\quad\quad\quad Key = SecureHash(TestKey)$
12 $\quad\quad\quad EndLoop$
13 Return Key ;

Le processus de l'algorithme 4 peut se résumer aux étapes suivantes :

1. **Qantification :**
   Une fois les valeurs CSI sont collectées, on passe à l'étape de quantification (ligne 4 dans Algorithme 4), en utilisant une nouvelle technique de démodulation QAM pour extraire les bits. Puisque les valeurs CSI sont des nombres complexes, nous considérons l'étape de quantification comme un problème de démodulation QAM, car la fonction normale du démodulateur QAM est de transformer chaque nombre complexe de données en une séquence de bits en fonction de sa valeur de bits correspondante dans le diagramme de constellation (Figure 5.3). Dans notre approche proposée, nous utilisons la même fonctionnalité, mais au lieu d'utiliser des données modulées par QAM, nous utilisons les valeurs CSI comme entrée. Plusieurs démodulations M-QAM d'ordre différents(4,16 et 64) ont été évaluées pour le quantificateur et nous avons observé que la démodulation 64-QAM donne des séquences de bits les plus aléatoires par rapport aux autres.

2. **Amplification de la confidentialité :** Une fois les valeurs quantifiées, nous appliquons un algorithme de hachage sécurisé aux séquences de bits extraites (ligne 11 dans Algorithme 4). Ceci est fait pour renforcer la propriété aléatoire et aussi pour éviter d'avoir des clés et des séquences répétées. De plus, la fonction de hachage peut fournir plus de secret aux clés générées.

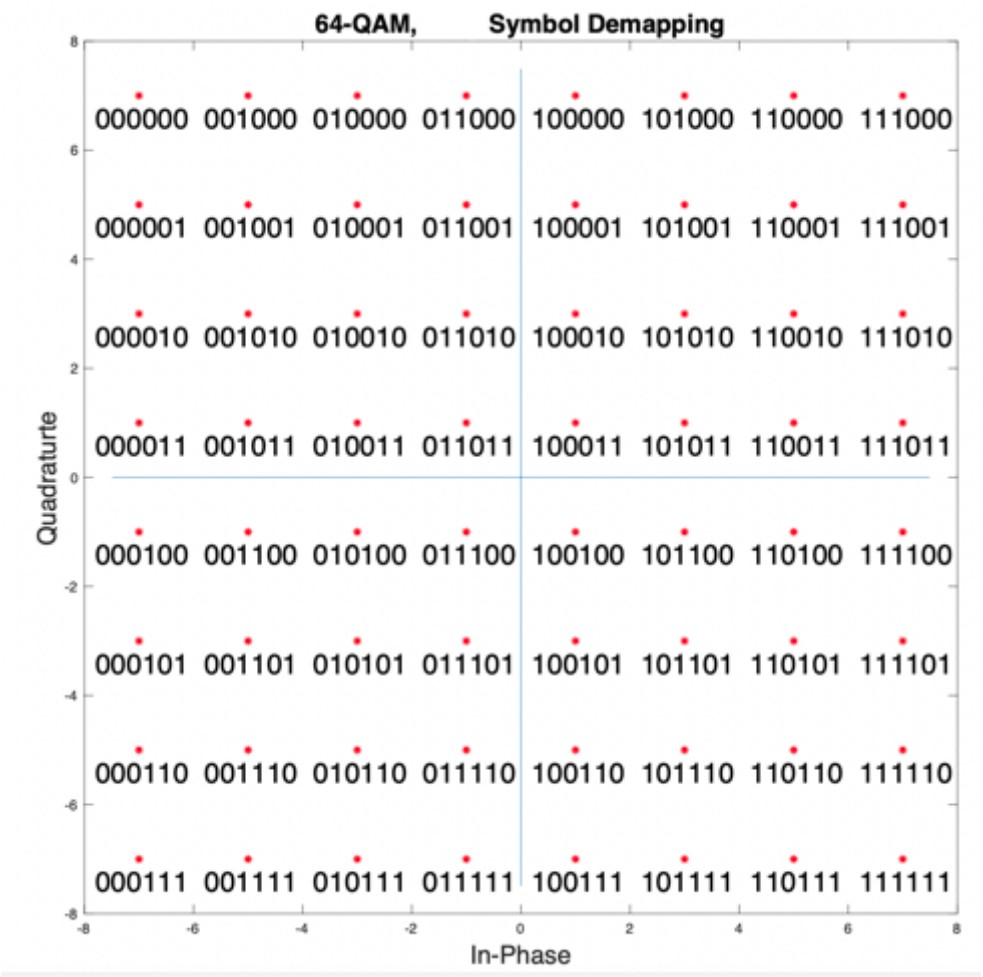


FIGURE 5.3: Diagramme de constellation de la 64-QAM.

5.5 Tests expérimentaux

Pour valider notre algorithme de génération de clés proposé et aussi afin d'assurer que ses performances correspondent aux exigences d'un environnement véhiculaire, nous avons réalisé des expériences dans des scénarios de conduite réels à Irvine (Californie) aux USA. Pour ces expériences, nous avons utilisé des cartes radio SDR de type Ettus USRP B210 embarqués dans des véhicules, qui constituent un réseau véhiculaire. Notre configuration de test consiste à utiliser deux pairs de communication half-duplex (émetteur/récepteur), où le Véhicule 1 (Alice) qui représente un véhicule connecté échange des messages à travers un réseau 802.11p, avec le Véhicule 2 (Bob) : un autre véhicule connecté ou une Road SideUnit (RSU). Cette configuration est visualisée dans la figure 5.4.
En utilisant GNURadio, le standard IEEE 802.11p est implémenté dans le logiciel et ensuite exécuté sur les cartes Ettus B210 pour produire des transmetteurs IEEE 802.11p [8].
Étant donné que les scénarios de conduite réels impliquent des facteurs qui varient constamment et ne peuvent pas être contrôlés, nous avons défini les scénarios des tests suivant la variation de la vitesse, qui nous a conduit à faire un scénario urbain/résidentiel (15 mph - 20 mph) et un scénario autoroutier (40 mph - 50 mph). Ces paramètres d'expérimentation sont communs aux tests RSSI et CSI.
Dans chaque scénario , les valeurs RSSI et CSI sont collectées à partir de chaque carte

FIGURE 5.4: Configuration expérimentale avec les cartes Ettus B210 embarquées dans les véhicules.

dans laquelle les clés de longueurs différentes sont générées en post-traitement : 16, 32, 64, 128.

5.5.1 Collecte des données et pré-traitement

5.5.1.1 Méthode basée sur les RSSI

Comme le montre la figure 5.5, les mesures RSSI sont calculées à partir des données brutes qui sont collectées à partir d'un enregistreur de données sur l'émetteur-récepteur. Le décodage de trame fait référence au processus par lequel les paquets sans fil sont reçus, synchronisés, et décodés. Les données brutes se présentent sous la forme de valeurs de puissance complexes qui sont traitées par les blocs fonctionnels suivants : Sync Short, Sync Long et Frame Equalizer. Les blocs Sync Short et Sync Long, sont respectivement responsables de la détection des trames de paquets potentielles et de l'alignement des trames. Ces blocs fonctionnels sont des blocs GNUradio synchrones, ce qui signifie que les données sont toujours reçues périodiquement par l'émetteur-récepteur, et ainsi, l'enregistreur de données est programmé pour n'échantillonner que des valeurs de puissance complexes lorsque Sync Short indique qu'il a détecté une trame.
À l'aide d'une fonction de mesure RSSI personnalisée, ces valeurs de puissance complexes sont converties en valeurs RSSI mesurées en décibel-milliwatts (dBm), produisant environ 14 valeurs RSSI par paquet reçu. Contrairement aux valeurs de puissance brutes, ces valeurs RSSI sont scalaires et peuvent maintenant être utilisées comme données d'entrée pour l'algorithme de génération de clés 3. Cependant, ce calcul des valeurs RSSI, introduit de la latence car il nécessite des filtres et des transformations en pré-traitement. Les clés sont générées en utilisant toutes les combinaisons possibles des paramètres : seuil α et la taille du groupe G_{size} indiqués dans le tableau suivant :

Alpha	0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9
G_{size}	20, 40, 60, 80, 100

TABLE 5.1: Paramètres de génération de clés pour l'algorithme basé sur RSSI.

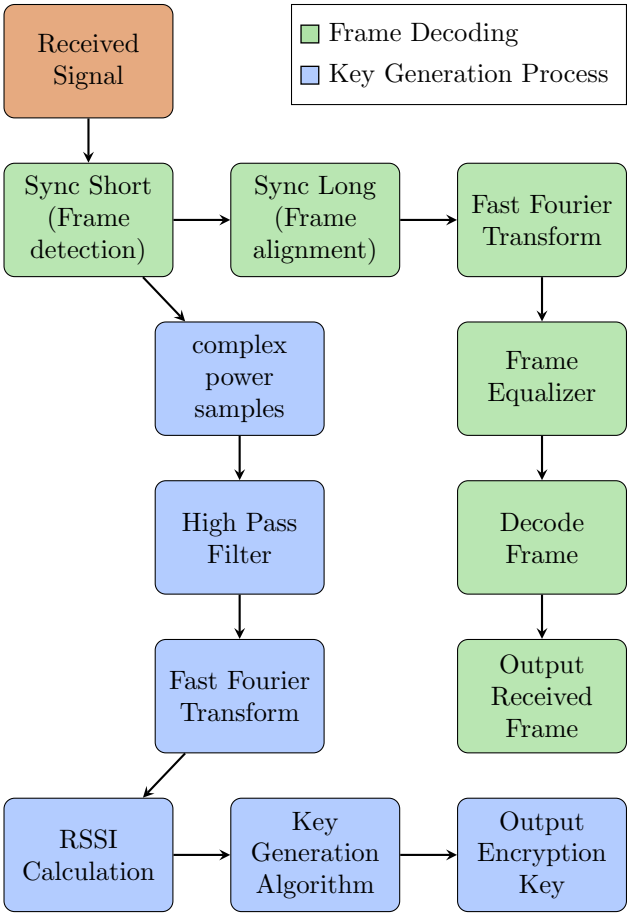


FIGURE 5.5: Organigramme de processus de génération de clés basé sur RSSI

Un sous-ensemble de 5000 échantillons à partir de 902 870 valeurs RSSI collectées dans un scénario de test de conduite réel est affiché à la figure 5.6. En se basant sur la forme et l'amplitude des résultats affichés, il est évident que les valeurs RSSI collectées au niveau du véhicule A et véhicule B seraient fortement corrélées si elles étaient alignées. Cette corrélation potentiellement élevée est due à la couche physique partagée ou au canal sans fil partagé entre le véhicule A et le véhicule B, ce qui signifie que la réponse du canal doit être similaire. Cependant, les retards causés par les multi-trajets sur le signal reçu peuvent affecter négativement cette corrélation. D'après [82], cette corrélation permet la génération de clés symétriques similaires. Des techniques de cross-corrélation ont été utilisées pour déterminer le retard provoqué par les différences de synchronisation de réception de paquets et pour décaler l'ensemble retardé de valeurs RSSI en conséquence. D'après le résultat affiché à la Figure 5.6, on constate que les valeurs RSSI ne sont pas totalement alignées.

5.5.1.2 Méthode basée sur les CSI

La figure 5.7, montre que les valeurs CSI sont collectées à la sortie du bloc fonctionnel Frame Equalizer, qui effectue l'étape d'égalisation du canal en estimant les valeurs CSI. Le décodage de trame est décrit plus haut dans la partie collecte des données RSSI. Il est important de noter que les valeurs CSI ne sont calculées que lorsqu'un paquet a été entièrement reçu contrairement aux valeurs RSSI issus du Sync Short qui produit parfois de faux positifs pour la réception des paquets. Compte tenu de cette propriété de la réception complète des paquets (où on a 64 valeurs CSI par paquet reçu), les valeurs CSI ne sont pas collectées à partir d'un paquet mal reçu.

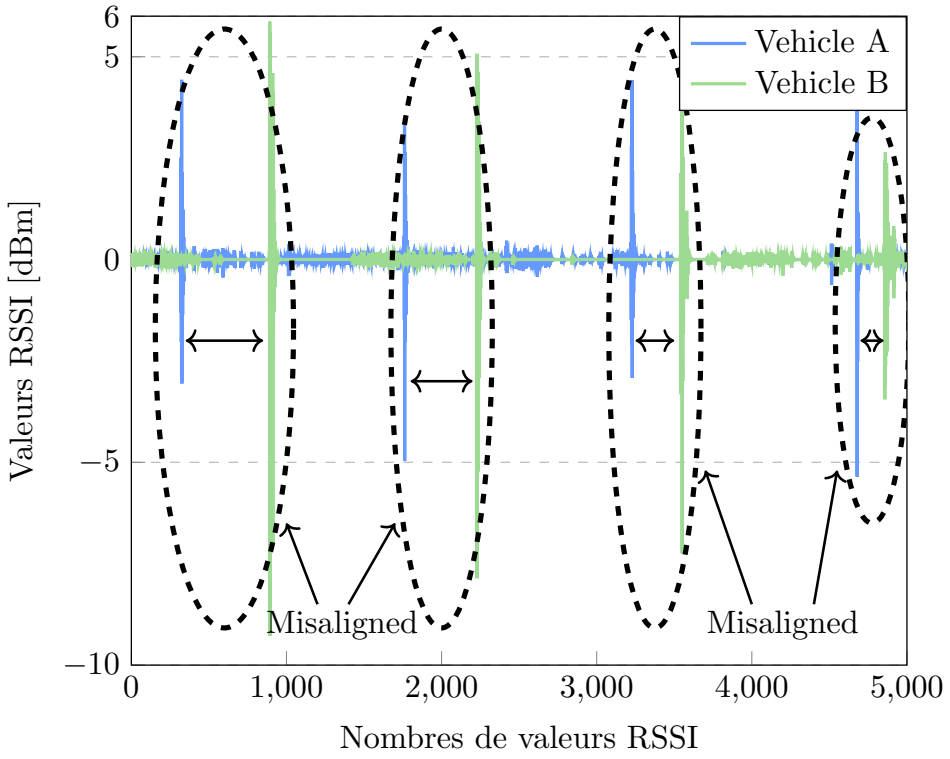


FIGURE 5.6: Valeurs RSSI collectées des véhicules A et B dans un scénario urbain.

En utilisant la quantification basée sur la démodulation QAM décrite dans la section 5.4.2, les valeurs CSI complexes peuvent être converties en bits pour générer des clés comme indiqué dans l'algorithme 4.
Un sous-ensemble de 256 échantillons provenant de 3264 valeurs CSI collectées lors d'un scénario de test de conduite réel est présenté à la figure 5.8. D'après la forme et l'amplitude des résultats affichés, on voit que les valeurs CSI collectées au véhicule A et au véhicule B sont fortement corrélées. Cette forte corrélation des valeurs CSI est probablement le résultat du canal de transmission partagé entre le véhicule A et le véhicule B, indiquant que les caractéristiques corrélées des canaux partagés sont représentées dans les estimations CSI. Par conséquent, étant donné que les valeurs CSI peuvent caractériser un canal de transmission comme les RSSI et tandis que les RSSI ont démontré un potentiel comme source d'entropie [82], nous pouvons conclure que les valeurs CSI sont une source d'entropie fiable pour générer des clés symétriques. L'alignement, dans ce cas, est obtenu sans avoir besoin d'utilisation des techniques de cross-corrélation ou similaires. Compte tenu de cet alignement réussi et de l'absence de mesures accidentelles à partir de paquet mal reçu, toutes les valeurs CSI sont produites par des paquets réels et peuvent servir comme données d'entrée pour l'algorithme 4. À 64 valeurs CSI par paquet, les estimations CSI peuvent être considérées comme une source d'entropie à rendement élevé.

5.5.2 Résultats et pourcentage de clés similaires pour les RSSI

Après avoir collecté les valeurs RSSI à partir de tests menés selon les scénarios de conduite réels décrits précédemment, les mesures RSSI sont utilisées par l'algorithme 3 pour générer des clés. Les clés sont générées séparément au véhicule A et au véhicule B indépendamment avant d'effectuer une étape de réconciliation.
La figure 5.9, représente le pourcentage de clés similaires générées aux véhicules A et B, pour chaque longueur de clé en appliquant toutes les combinaisons de pa-

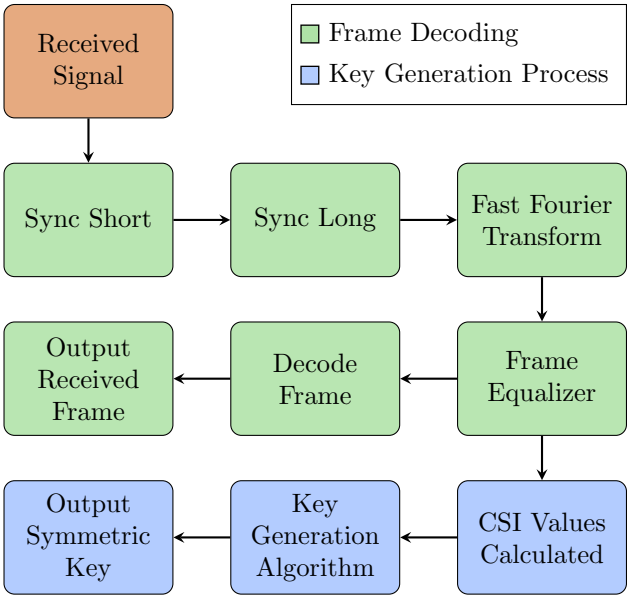


FIGURE 5.7: Organigramme décrivant le processus de génération de clés basé sur CSI

ramètres possibles. La proportion de clés similaires pour toutes les longueurs de clé est inférieure à 10% et démontre une relation inversement proportionnelle avec la longueur de clé, ce qui signifie que le taux de similarité diminue à mesure que la longueur de clé augmente. Cependant, ce taux d'appariement relativement faible n'est possible que dans le cas où les mesures RSSI corrélées sont alignées. Si ces mesures ne sont pas alignées, comme indiqué dans le taux de similarité pour un alignement infructueux, il y a 0% de clés similaire. Pour ces tests, un alignement réussi n'a été obtenu que dans le scénario urbain plus statique à des vitesses faibles de 15 à 20mi/h.

Longueur Clé	Clé Générée
16	0010000000000111
32	00000000001111111111111110010000
64	0000000000111111111111111001000000000001111111111111111001000 00000
128	0000000000111111111111111001000000000001111111111111111001000 0000000111111111111111001000000000001111111111111111001000000 0000111111
256	0000000000111111111111111001000000000001111111111111111001000 0000000111111111111111001000000000001111111111111111001000000 0000111111111111111001000000000001111111111111111001000000000 0111111111111111001000000000001111111111111111001000000000011 1111111111110010000

TABLE 5.2: Clés générées à partir de mesures RSSI dans les scénarios de test urbains

Des exemples de clés similaires avec une longueur variable obtenue dans ce cas peuvent

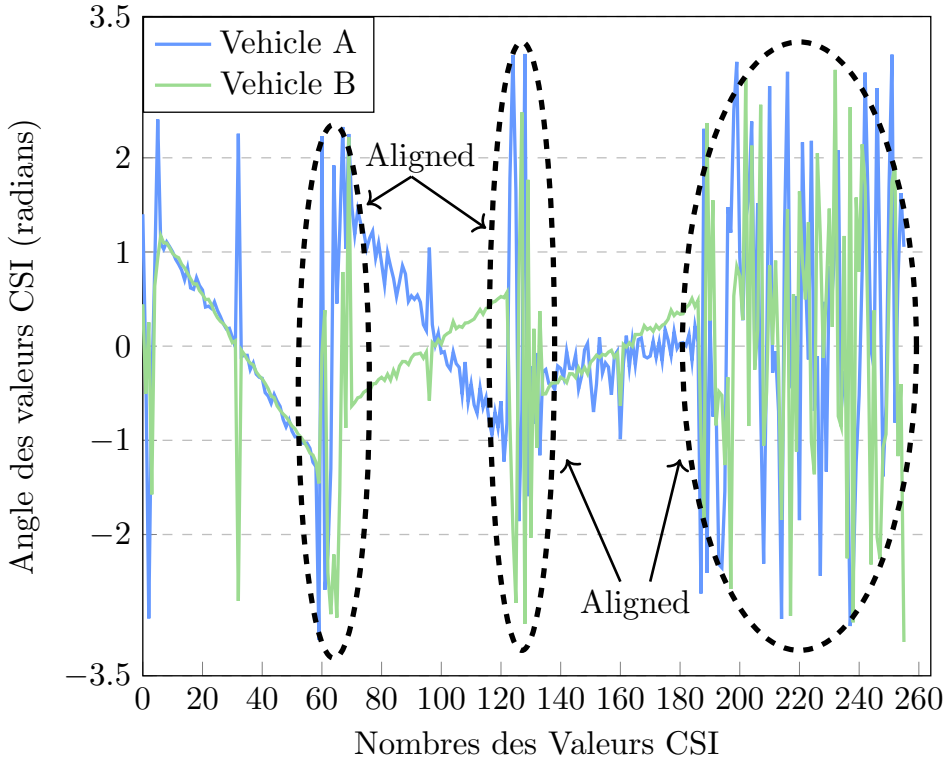


FIGURE 5.8: Valeurs CSI collectées des véhicules A et B dans un scénario autoroute.

être trouvés dans le tableau 5.2. Néanmoins, ces clés similaires n'ont été produites que dans des conditions plus statiques, ce qui ne peut pas toujours être prévu dans les scénarios d'utilisation V2X normaux. À des vitesses de 40 à 50 mi/h dans le scénario autoroutier dynamique, l'alignement du signal n'a pas pu être obtenu même en post-traitement, et par conséquent, le taux de similarité est de 0%. Compte tenu de la corrélation apparente entre le taux de similarité et l'alignement, il s'avère donc nécessaire d'aligner les mesures RSSI.

5.5.3 Résultats et pourcentage de clés similaires pour les CSI

Une fois que les valeurs CSI ont été collectées à partir de tests effectués selon les scénarios de conduite, les valeurs CSI sont exploitées par l'algorithme 4, pour générer des clés. Les clés sont générées séparément dans chaque véhicule (véhicules A et B). La figure 5.10, représente le pourcentage de clés similaires générées pour les deux scénarios de tests : urbain à faible vitesse et autoroute à grande vitesse. Pour le scénario urbain plus statique avec des vitesses qui varient de 15 à 20 mi/h, la longueur de clé et le taux de similarité ont une relation proportionnelle, de sorte que le taux de similarité augmente lorsque la longueur de clé est plus grande. L'inverse est vrai pour le scénario autoroutier dynamique avec des vitesses plus élevées de 40 à 50 mi/h, où le taux de similarité diminue à mesure que la longueur de clé et plus importante.
Comme décrit dans la section 5.4.2, notre méthode de génération de clés consiste à quantifier les valeurs CSI reçues en un flux de bits qui est utilisé pour produire la clé demandée. Chaque ensemble de clés dans les deux scénarios sont généré à partir d'un seul ensemble de données, ce qui signifie que les clés de chaque scénario sont générées à partir des données finies collectées dans ce test. Par conséquent, la relation proportionnelle entre le taux de similarité et la longueur de clés pour le scénario urbain n'est probablement pas le résultat d'une augmentation des bits similaires, mais d'avoir les mêmes bits similaires pour le moins de clés générées à des longueurs de clé plus

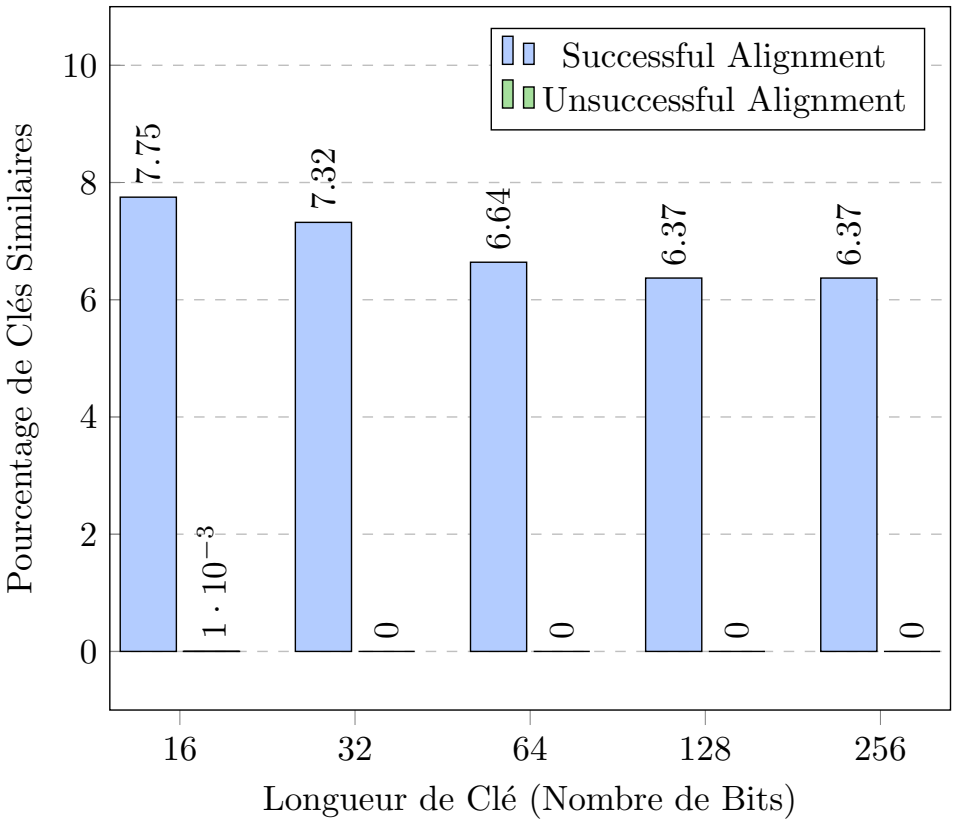


FIGURE 5.9: Taux de clés similaires générées pour les véhicules A et B pour un alignement réussi et infructueux.

longues. Cependant, dans le scénario autoroutier, l'environnement est plus dynamique et peut entraîner des paquets perdus ou des interruptions dans la transmission. Étant donné que les CSI estimés nécessitent des paquets entièrement reçus, l'instabilité du canal empêche la génération de clés plus longues et se traduit par un taux de similarité de clé plus faible pour des longueurs de clé plus grandes, comme indiqué sur la Figure 5.10.

Des exemples de clés générées dans le scénario autoroutier dynamique sont affichés dans le Tableau 5.3, ce qui prouve que les valeurs CSI peuvent être utilisées comme source d'entropie fiable, qui répond aux exigences de l'environnement véhiculaire.

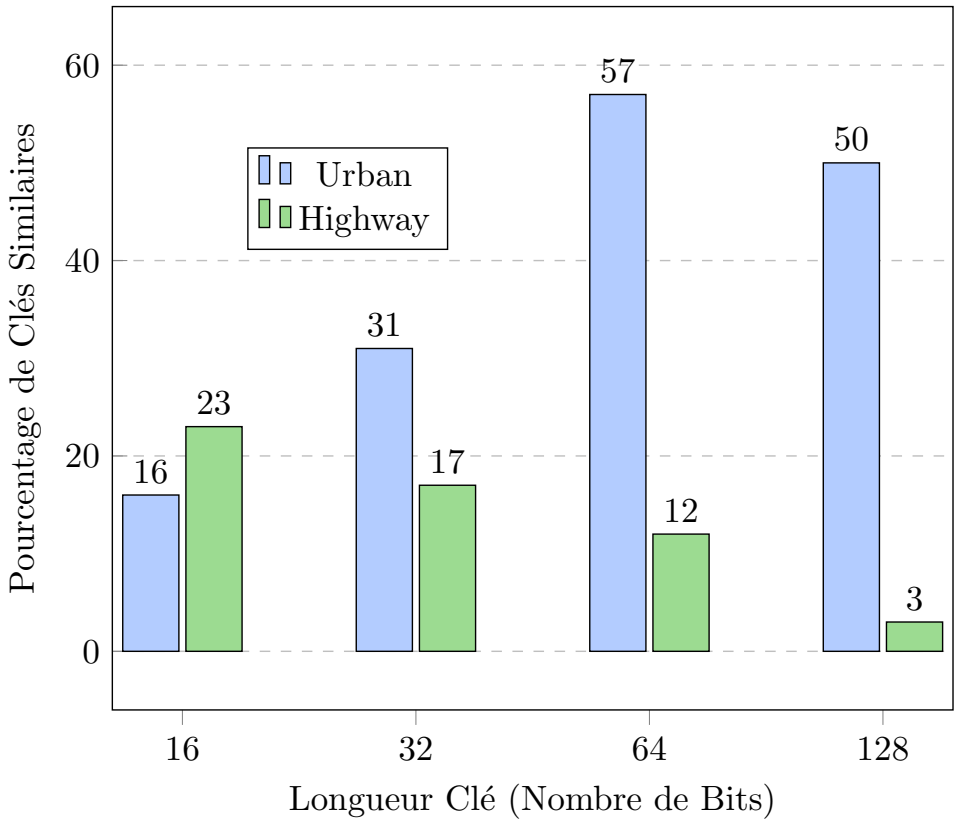


FIGURE 5.10: Taux de similarité pour les scénarios urbains et autoroutiers.

Longueur Clé	Clé Généré
16	1100011110010110
32	11011011001111001101101101100101
64	1100101100011110010110100110010111010111001011 100011001001110011
128	1100100110010011000011100100110001111100011000 1110010110100110101110001111000011001011100111 0100110010111010011100011000011110111

TABLE 5.3: Clés générées à partir des CSI estimées dans les scénarios autoroutiers

5.6 Évaluation des performances de l'approche CSI

Afin de pouvoir évaluer les performances de notre méthode de génération de clés proposée, nous avons calculé les métriques suivantes :

— **Min-entropie**

— **Temps de génération de clés**

— **complexité de l'algorithme**

5.6.1 Min-entropie

Le concept de min-entropie est utilisé pour évaluer la propriété aléatoire et la sécurité d'une clé, qui consiste à estimer l'entropie du cas le plus défavorable qui fournit une borne inférieure sur la propriété aléatoire de la clé. Si K est l'ensemble de toutes les clés générées aléatoirement possibles, l'équation pour le min entropie est la suivante :

$$H_\infty = H_{min} = -\log(\max_{k \in K} \Pr[k = K]) \tag{5.1}$$

$\Pr[k = K]$, représente la probabilité de générer une clé $k \in K$. En se basant sur cela, nous décrivons niveau de sécurité comme suit :

$$Security_{str} = H_{min}/Key_{size} \tag{5.2}$$

Key_{size} est la longueur de la clé et $Security_{str}$ est une valeur comprise entre 0 et 1. Plus la valeur est grande, plus le niveau de sécurité est élevé car elle fournit plus de bits d'entropie.
Pour estimer le min-entropie de notre algorithme proposé (Algorithme 4), nous utilisons l'équation 5.1. Nous générons d'abord 46352 clés à partir des valeurs CSI collectées dans le test du scénario autoroutier pour former un ensemble de clés K qui est supérieure ou égal à 25600 clés, qui est le nombre de clés nécessaires pour conclure raisonnablement que le calcul produit un min-entropie moyen. À partir de cet ensemble de clés, la clé avec l'apparence la plus fréquente est utilisée pour générer Pr_{max} ou $Pr[k = K]$. Notre algorithme de génération de clés proposé a un min-entropie moyen de **70,52 %**.

5.6.2 Temps de génération de clés

Étant donné que notre algorithme de génération de clés proposé est destiné à être utilisé dans un environnement critique, il est nécessaire que le temps de génération de clés réponde systématiquement aux exigences de latence dans chaque scénario. D'après [64], cette limite de temps se situe généralement entre 50 et 200 millisecondes. Le tableau 5.4, représente le temps moyen de génération de clés pour les différentes longueurs.
Notre méthode de génération de clés basée sur le CSI a donné des temps d'exécution à l'échelle de **microsecondes**, bien en dessous de la plage exigée de 50 à 200 ms comme on le voit dans les résultats du tableau 5.4. La source de cette bonne performance est probablement l'utilisation des CSI estimés, dont on a 64 valeurs utilisables par paquet et aussi due à l'utilisation de la démodulation QAM à faible latence qui quantifie ces valeurs en clés de 64 bits ou plus en microsecondes.
En raison de ces mécanismes, notre proposition a une faible surcharge dans tous les scénarios car elle ne nécessite que quelques paquets pour la génération de clés et utilise efficacement toutes les données collectées.

Longueur de Clé	Temps d'exécution moyen (μs)
16	18.65
32	34.75
64	78.45
128	149.58

TABLE 5.4: Temps d'exécution moyen pour la génération de clés.

5.6.3 complexité de l'algorithme

Nous avons évalué la complexité Big-O de notre algorithme proposé en terme de taille d'entrée (CSIset) et nous avons constaté que notre algorithme a un temps d'exécution dans le pire des cas de l'ordre de $O(nlogn)$. Ainsi, nous pouvons dire que notre algorithme est un algorithme super-linéaire où le temps d'exécution augmente approximativement proportionnellement à la taille du CSIset. Cela correspond à nos résultats expérimentaux présentés dans le Tableau 5.4.

Afin de déterminer le degré de sécurité de notre méthode proposée, face à des attaques de pirates, nous avons évalué notre algorithme dans des scénarios d'attaques. Dans ce travail on a considéré deux types d'attaques :

— **Attaque non-intrusive**

— **Attaque de brute force**

5.6.4 Attaque non-intrusive

Dans ce type d'attaque, on considère un attaquant (Eve), qui tente d'écouter la communication entre les deux parties légitimes (Alice et Bob) via un troisième canal, comme le montre la figure 5.11. On suppose qu'Eve peut capturer tous les paquets sans fil et connaît les caractéristiques du réseau V2X. Dans ce cas, si Eve peut obtenir la même clé symétrique, alors le système est considéré comme vulnérable. Les résultats démontrent qu'un attaquant situé à différentes positions par rapport

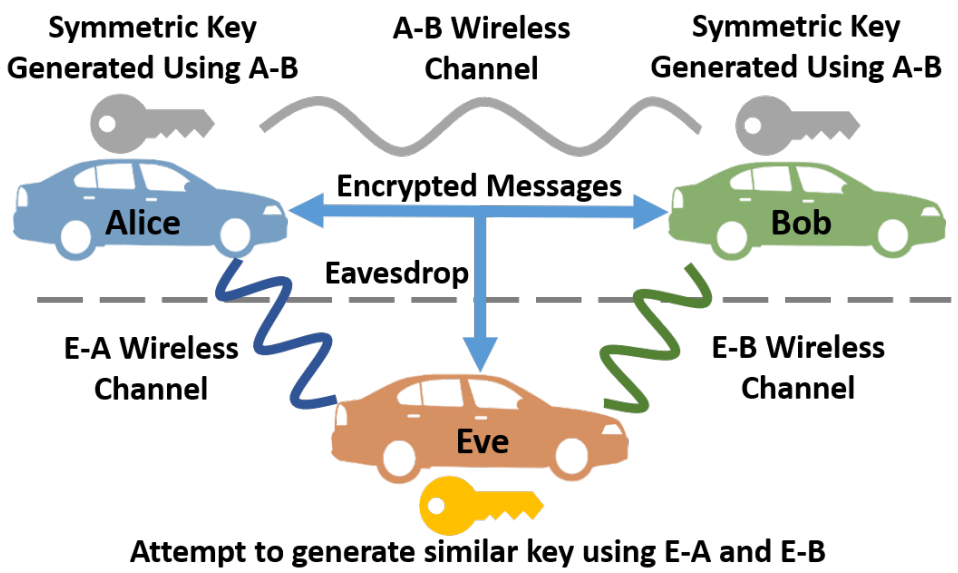


FIGURE 5.11: Attaque non-intrusive.

Longueur de Clé	Taux de discordance pour l'attaquant E (%)
16	52.38
32	52.60
64	50.97
128	54.29

TABLE 5.5: Taux de discordance de bits moyen pour les clés générées par le véhicule attaquant E

aux véhicules légitimes mesure des caractéristiques de canal sans fil différentes et ne peut pas générer la même clé secrète que les parties du canal légitime. Selon les

résultats présentés dans le tableau, il est clair que les clés générées à partir du canal entre le véhicule A et le véhicule E ont un taux de discordance supérieur à 50% avec les clés générées à partir du canal entre le véhicule A et le véhicule B. Cela répond à l'unicité requise et renforce le principe de base de notre méthode. Où, un attaquant situé à plusieurs positions par rapport aux véhicules légitimes, ne sera pas en mesure de prédire/générer la clé secrète en raison de différences dans les caractéristiques de canal partagé.

5.6.5 Attaque de brute force

Le temps de brute force est le temps maximum nécessaire à un attaquant pour casser une clé secrète en essayant toutes les combinaisons possibles. Une clé de longueur k a (2^k) valeurs possibles. Plus la longueur de la clé est grande, plus le temps de brute force est long et plus la clé est forte. Le temps de force brute dépend de la longueur de la clé et des capacités de l'attaquant (opérations en virgule flottante par seconde). En conséquence, des longueurs de clé plus longues sont généralement plus préférables car elles sont plus difficiles à forcer.

taille de clé (bits)	Temps à 10^6 FLOPs	Temps à 10^9 FLOPs	Temps à 10^{17} FLOPs (Taihu-Light)	Temps à 10^{18} FLOPs (quantum computer)	Période de validité
16	65 ms	65 μs	$6.5X10^{-4}$ ns	$6.5X10^{-5}$ ns	1 message
32	1.17 Heurs	4.2 s	0.42 ns	$0.42X10^{-1}$ ns	1 message
64	$75.03X10^4$ années	570.39 années	$1.8X10^2$ s	18 s	1 minute
128	$1.08X10^{25}$ années	$1.08X10^{22}$ années	$1.08X10^{14}$ années	$1.08X10^{13}$ années	15 minutes

Table 5.6: Temps Brute Force vs longueur de clé.

Comme le montre le tableau 5.6, le temps de brute force a été évalué pour différentes tailles de clé et différents niveaux de capacités d'attaquant, y compris ceux des super-ordinateurs (TaihuLight) et des ordinateurs quantiques.
Puisque notre travail se concentre sur des attaques pratiques et réelles dans un environnement véhiculaire, une attaque par brute force en utilisant un super-calculateur TaihuLight ou un ordinateur quantique est extrêmement improbable. Cependant, nous les présentons ici pour démontrer que nos longueurs de clé et nos temps de rotation (période de validité) garantissent la sécurité avec le matériel informatique conventionnel et futur. D'après le tableau 5.6, il est clair que les clés de moins de 64 bits ne sont pas très sécurisées contre les attaques par force brute. Cependant, nous proposons la rotation des clés 16 et 32 bits pour chaque message afin d'éviter la possi-

bilité qu'un attaquant utilise la clé cassée pour se faire passer pour Alice ou Bob. De plus, les informations avec une confidentialité importante peuvent être réservées pour une utilisation avec des clés de 128 bits, tandis que des tailles de clés plus petites sont utilisées pour des messages immédiats qui ne contiennent pas de données sensibles.

5.6.6 Classification du générateur de bits aléatoires suivant les standards de NIST

Selon les recommandations [6] de l'Institut national des normes et de la technologie (NIST), il existe deux classes de générateurs de bits aléatoires (RBG). La première classe utilise du matériel dédié ou des expériences physiques pour générer des bits aléatoires, où chaque bit de sortie est basé sur un processus physique imprévisible. Les méthodes de cette classe sont appelées générateurs de bits aléatoires non déterministes (NRBG). La deuxième classe comprend des méthodes qui calculent des séquences de bits de manière déterministe sur la base de méthodes de génération de nombres pseudo-aléatoires utilisant des algorithmes spécifiques. Les méthodes de cette classe sont appelées générateurs de bits aléatoires déterministes (DRBG).
Ainsi, notre approche proposée peut être classée comme NRBG. À partir des recommandations du NIST [77], le modèle de source d'entropie pour les NRBG est illustré à la figure 5.12.

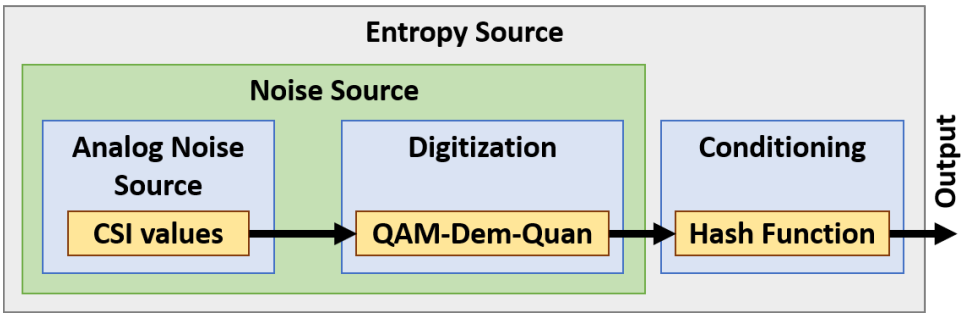


FIGURE 5.12: source d'entropie

Le bloc source d'entropie comprend les composants suivants :

— **Source de bruit :** C'est la racine de la sécurité pour la source d'entropie et le RBG dans son ensemble. Il fournit les séquences non déterministes (valeurs CSI dans notre approche) du processus physique qui est le canal sans fil véhiculaire dans la configuration de nos expériences. Comme le montre la figure, le processus d'échantillonnage comprend la numérisation pour convertir le bruit analogique en données binaires. Dans notre méthode proposée, comme les valeurs CSI sont des nombres complexes, la numérisation est effectuée par le quantificateur démodulateur QAM (QAM-Dem-Quan)

— **Conditionnement optionnel :** Ce composant vise à augmenter l'entropie des bits de sortie résultants. Dans notre approche, nous utilisons une fonction de hachage comme composant de conditionnement comme indiqué dans l'algorithme 4.

5.6.7 Comparaison avec l'état de l'art

Le degré de sécurité des clés générées par l'algorithme proposé est comparé à celles produites par d'autres techniques. Nous avons utilisé le min-entropie moyen comme indicateur de performance clé (KPI).
Les clés pré-distribuées et les algorithmes associés sont prévisibles, les approches de clé pré-distribuées ont peu ou pas d'entropie. Les approches qui utilisent les PUF, comme le SRAM-PUF [31], peuvent générer des clés avec une entropie minimale moyenne élevée.
La Figure 5.13 montre le min-entropie moyen résultant de notre technique proposée, comparé avec le min-entropie d'autres techniques bien connues telles que les clés pré-distribuées, DFF-PUF, SRAM-PUF et aussi avec l'approche qui exploite les mesures RSSI (RSSI-PHY). On voit que notre approche basée sur les valeurs CSI a un min-entropie de 70,52%, qui est supérieur à celui de RSSI-PHY qui a un min entropie autour de 67%.
À noter que notre algorithme peut générer des clés avec un niveau de sécurité largement supérieure que celui de certaines approches basées sur PUF (DFF-PUF qui a un min-entropie de 50%). Bien que certaines des approches basées sur PUF (par exemple, SRAM-PUF) puissent générer des clés avec une entropie minimale moyenne plus élevée (puisque le nombre de bits 0 et 1 ont tendance à être à peu près identiques), notre algorithme a l'avantage de générer des clés sans avoir besoin d'un processus physique spécial tel que le redémarrage de la SRAM (pour SRAM-PUF). Aussi le min-entropie moyen (70%) peut être potentiellement amélioré en ajoutant des améliorations matérielles.

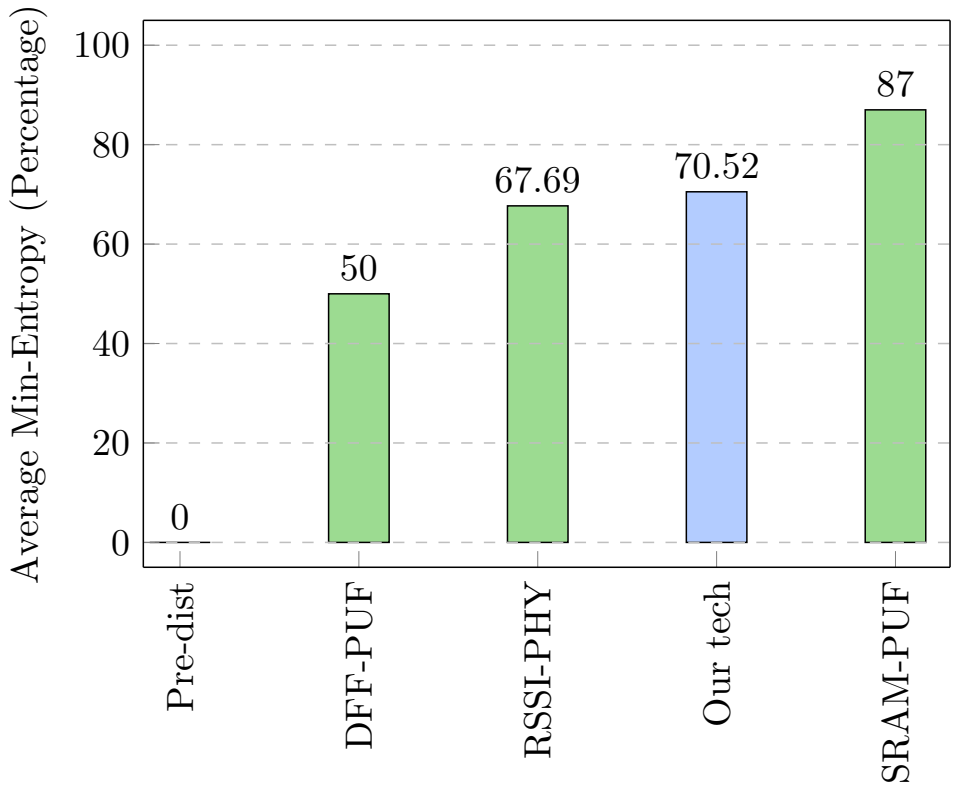


FIGURE 5.13: Comparaison de notre min-entropie moyen avec l'état de l'art

5.7 Conclusion

Dans ce dernier chapitre, nous avons présenté une technique de génération de clés à base de la couche physique, en exploitant le caractère aléatoire du canal sans fil sous la forme des CSI estimées. Cette approche peut générer des clés symétriques secrètes pour sécuriser les communications véhiculaires. Notre méthode résout le défi de la sécurité consistant à préserver la confidentialité des communications V2X et peut-être considérer comme étant une source d'entropie fiable pour générer des clés à partir de canaux sans fil V2X.
Les résultats de nos tests dans un environnement réel ont démontré que notre méthode a un taux de génération de clés mesurées en microsecondes, bien dans la plage opérationnelle exigée dans divers scénarios de communication V2X. Ces résultats ont également montré que les clés générées ont un min-entropie moyen de 70,52 % et ont donc un degré de sécurité comparable aux méthodes existantes dans l'état de l'art. En résumé, nous avons validé notre technique de génération de clés basée sur CSI comme une solution pratique pour sécuriser les communications véhiculaires.

Chapitre 6

Conclusion générale

6.1 Conclusion

Dans ce travail nous avons présenté nos travaux de recherche qui ont été concentrés essentiellement autour des trois axes suivants :

— Dans le premier axe, nous nous sommes intéressés à l'estimation de canal pour les systèmes OFDM dans un environnement véhiculaire. Nous avons proposé un nouvel algorithme (MSCR) qui se base sur l'utilisation de la séquence d'apprentissage connue dans les trames pour estimer les CSI. Cet algorithme a été évalué sur un canal véhiculaire modélisé sous Matlab en se basant sur des mesures environnementales des canaux véhiculaires réels.
Les résultats ont montré que notre approche basée sur l'algorithme MSCR convient aux exigences des communications V2X, en donnant un SER faible et bon temps de latence, dans un environnement véhiculaire sévère.

— Dans le deuxième axe, nous nous sommes intéressés à l'identification de l'environnement véhiculaire, dans lequel le véhicule circule. Nous avons proposé une nouvelle approche basée sur le Deep Learning, ou nous avons élaboré une architecture de réseaux de neurones CNN, qui utilise les CSI estimées, comme caractéristiques d'entrée pour entraîner et tester le modèle.
Les résultats obtenus ont démontré que notre méthode peut atteindre une précision d'identification de toutes les types d'environnements véhiculaires, autour de $96,48\%$. Les résultats présentés indiquent que le temps de prédiction mesurée pour notre modèle CNN est en microsecondes, ce qui convient aux exigences des divers scénarios de la conduite autonome.

— La sécurité des communications véhiculaires est la thématique sur laquelle s'est déroulé notre dernier axe de recherche, ou nous avons présenté une nouvelle technique de génération de clés à base de la couche physique. Cette approche consiste à utiliser les caractéristiques du canal sans fil sous la forme des CSI estimées, comme source aléatoire, pour générer des clés symétriques.
L'algorithme proposé a été implémenté sur des cartes USRP-B210 et les résultats de tests expérimentaux réalisés ont démontré que le min-entropie des clés générées est autour de $70,52\%$, ce qui représente un degré de sécurité important et comparable aux méthodes existantes. En plus notre algorithme peut garantir un temps de latence de générations de clés dans la plage opérationnelle exigée dans les recommandations des communications V2X, ce temps est mesuré autour des microsecondes. Finalement notre approche peut être validée comme étant solution pratique pour sécuriser les communications véhiculaires.

6.2 Perspectives

Comme perspectives à ce travail, Nous pouvons citer :

— Implémenter l'algorithme MSCR d'estimation de canal sur des cartes SDR, afin d'évaluer ses performances dans un environnement véhiculaire réel.

— Proposer une nouvelle version de l'algorithme MSCR adapté aux technologies de communication cellulaires (5G et future 6G) qui peuvent supporter des communications MIMO.

— Implémenter le modèle abordé pour l'identification de l'environnement sur des cartes USRP afin d'évaluer ses performances dans un environnement réel.

— Proposer une nouvelle approche d'estimation de canal en utilisant des méthodes du Deep Learning en se basant sur l'identification de l'environnement.

— Étudier la validité et la faisabilité de l'approche de génération de clés basée sur les CSI, dans un environnement à plus grande échelle avec plus de véhicules.

Acronymes

3GPP 3rd Generation Partnership Project.

AI Artificial Intelligence.
ANN Artificial Neural Network.

BER Bit Error Rate.

CFR Channel Frequency Response.
CI Confidence Interval.
CNN Convolutional Neural Networks.
CRC Cyclic Redundancy Check.
CSI Channel State Information.
C-V2X Cellular V2X.

DBN Deep Belief Networks.
DL Down-Link.
DNN Deep Neural Networks.
DRBG Deterministic Random Bit Generator.

E-UTRAN Evolved UMTS Terrestrial Radio Access Network.
eMBB Enhanced Mobile Broadband.

GNB Gaussian Naive Bayes.

HSS Home Subscriber Server.

ICE Iterative Channel Estimation.
ISI InterSymbol Interference.
ITS Intelligent Transportation System.

KNN K-Nearest Neighbors.

LS Least Square.
LTE Long Term Evolution.
LTS Long Training Sequence.

MAC Media Access Control.
MANET Mobile Ad hoc Network.
MME Mobility Management Entity.
MMSE Minimum Mean Square Error.

MSCR Multiple Sequential Constraint Removal.
MSE Mean Squared Error.

NHTSA National Highway Traffic Safety Administration.
NIST National Institute of Standards and Technology.
NRBG Non-deterministic Random Bit Generator.
NR-V2X New Radio V2X.

OBU On Board Unit.
OFDM Orthogonal Frequency-Division Multiplexing.

PGW Packet Data Network Gateway.
PHY Physical Layer.
PKI Public key infrastructurs.
PM Parametric Model.

RBG Random Bit Generator.
RF Random Forest.
RNN Recurrent Neural Networks.
RSSI Received Signal Strength Indicator.
RSU Road Side Unit.

SCMS Security Credential Management System.
SCR Sequential Constraint Removal.
SDR Software-Defined Radio.
SER Symbols Error Rate.
SGW Serving Gateway.
SNR Signal-to-Noise Ratio.
STS Short Training Sequence.
SVM Support Vector Machine.

UE Unit Equipment.
UL Up-Link.
URLLC Ultra-Reliable Low-Latency Communications.
USRP Universal Software Radio Peripheral.

V2I Vehicle-To-Infrastructure.
V2V Vehicle-To-Vehicle.
V2X Vehicle-To-Everything.
VANET Vehicular Ad hoc Network.

Bibliographie

[1] Martín Abadi, Paul Barham, Jianmin Chen, Zhifeng Chen, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Geoffrey Irving, Michael Isard, et al. Tensorflow : A system for large-scale machine learning. In *12th {USENIX} symposium on operating systems design and implementation ({OSDI} 16)*, pages 265–283, 2016. 67

[2] Ryma Abassi. VANET security and forensics : Challenges and opportunities. *Wiley Interdisciplinary Reviews : Forensic Science*, 1(2) :e1324, March 2019. 84

[3] Paul Alexander, David Haley, and Alex Grant. Cooperative intelligent transport systems : 5.9-ghz field trials. *Proceedings of the IEEE*, 99(7) :1213–1235, 2011. 43, 52, 68

[4] Qazi Ejaz Ali, Naveed Ahmad, Abdul Haseeb Malik, Gauhar Ali, and Waheed Ur Rehman. Issues, Challenges, and Research Opportunities in Intelligent Transport System for Security and Privacy. *Applied Sciences*, 8(10) :1964, October 2018. 84

[5] Gunther Auer and Jérôme Bonnet. Threshold controlled iterative channel estimation for coded ofdm. In *2007 IEEE 65th Vehicular Technology Conference-VTC2007-Spring*, pages 1737–1741. IEEE, 2007. 48, 49

[6] Elaine Barker and John Kelsey. Nist special publication 800-90a recommendation for random number generation using deterministic random bit generators. 2012. 102

[7] Paulo SLM Barreto, Jefferson E Ricardini, Marcos A Simplício Jr, and Harsh Kupwade Patil. qscms : Post-quantum certificate provisioning process for v2x. *IACR Cryptology ePrint Archive*, 2018 :1247, 2018. 86

[8] Bastian Bloessl. *A Physical Layer Experimentation Framework for Automotive WLAN*. PhD thesis, Universitätsbibliothek, 2018. 90

[9] M. Boban, A. Kousaridas, K. Manolakis, J. Eichinger, and W. Xu. Connected Roads of the Future : Use Cases, Requirements, and Design Considerations for Vehicle-to-Everything Communications. *IEEE Vehicular Technology Magazine*, 13(3) :110–123, September 2018. 85

[10] Mate Boban, Apostolos Kousaridas, Konstantinos Manolakis, Josef Eichinger, and Wen Xu. Connected roads of the future : Use cases, requirements, and design considerations for vehicle-to-everything communications. *IEEE vehicular technology magazine*, 13(3) :110–123, 2018. 40, 55

[11] Jerome Bonnet and Gunther Auer. Optimized iterative channel estimation for ofdm. In *IEEE Vehicular Technology Conference*, pages 1–5. IEEE, 2006. 48, 49

[12] Benedikt Brecht, Dean Therriault, André Weimerskirch, William Whyte, Virendra Kumar, Thorsten Hehn, and Roy Goudy. A security credential management system for v2x communications. *IEEE Transactions on Intelligent Transportation Systems*, 19(12) :3850–3871, 2018. 85

[13] c Road. Radio frequencies designated for enhanced road safety in europe - c-roads position on the usage of the 5.9 ghz band. www.c-roads.eu, 2016. 19

[14] Fernando Camacho, César Cárdenas, and David Muñoz. Emerging technologies and research challenges for intelligent transportation systems : 5g, HetNets, and SDN. *International Journal on Interactive Design and Manufacturing (IJIDeM)*, 12(1) :327–335, February 2018. 84

[15] Chenyi Chen, Ari Seff, Alain Kornhauser, and Jianxiong Xiao. Deepdriving : Learning affordance for direct perception in autonomous driving. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2722–2730, 2015. 60

[16] François Chollet. Xception : Deep learning with depthwise separable convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1251–1258, 2017. 79

[17] Sinem Coleri, Mustafa Ergen, Anuj Puri, and Ahmad Bahai. Channel estimation techniques based on pilot arrangement in ofdm systems. *IEEE Transactions on broadcasting*, 48(3) :223–229, 2002. 46

[18] IEEE LAN/MAN Standards Committee et al. Ieee 802.11-wireless lan medium access control (mac) and physical layer (phy) specifications. In *IEEE, June*, 2007. 48

[19] Vinayak V Dixit, Sai Chand, and Divya J Nair. Autonomous vehicles : disengagements, accidents and reaction times. *PLoS one*, 11(12) :e0168054, 2016. 82

[20] Carl Ellison and Bruce Schneier. Ten risks of pki : What you're not being told about public key infrastructure. *Comput Secur J*, 16(1) :1–7, 2000. 85

[21] Mohamed Elwekeil, Taotao Wang, and Shengli Zhang. Deep learning for environment identification in vehicular networks. *IEEE Wireless Communications Letters*, 9(5) :576–580, 2019. 80, 81

[22] TCITS ETSI. Intelligent transport systems (its) ; access layer specification for intelligent transport systems operating in the 5 ghz frequency band. *EN 302 663 V1. 2.1, July 2013*. 7, 25

[23] TR ETSI. 103 257-1 v1. 1.1 (2019-05) intelligent transport systems (its). *Access Layer*. 44

[24] Ali Zarei Ghanavati, Udit Pareek, Sami Muhaidat, and Daniel Lee. On the performance of imperfect channel estimation for vehicular ad-hoc networks. In *2010 IEEE 72nd vehicular technology conference-fall*, pages 1–5. IEEE, 2010. 40

[25] Marco Giordani, Andrea Zanella, Takamasa Higuchi, Onur Altintas, and Michele Zorzi. Emerging Trends in Vehicular Communication Networks. In Karm Veer Arya, Robin Singh Bhadoria, and Narendra S. Chaudhari, editors, *Emerging Wireless Communication and Network Technologies : Principle, Paradigm and Performance*, pages 37–57. Springer Singapore, Singapore, 2018. 84

[26] Ian Goodfellow, Y Bengio, and A Courville. Machine learning basics. In *Deep learning*, volume 1, pages 98–164. MIT press, 2016. 61

[27] Farah Haidar, Arnaud Kaiser, and Brigitte Lonc. On the performance evaluation of vehicular pki protocol for v2x communications security. In *2017 IEEE 86th Vehicular Technology Conference (VTC-Fall)*, pages 1–5. IEEE, 2017. 85

[28] Hamssa Hasrouny, Abed Ellatif Samhat, Carole Bassil, and Anis Laouiti. VANet security challenges and solutions : A survey. *Vehicular Communications*, 7 :7–20, January 2017. 84

[29] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Identity mappings in deep residual networks. 2016. *arXiv preprint arXiv :1603.05027*, 2016. 79

[30] Hussein Hijazi. *Estimation de canal radio-mobile à évolution rapide dans les systèmes à modulation OFMD*. PhD thesis, Institut National Polytechnique de Grenoble-INPG, 2008. 43

[31] Daniel E Holcomb, Wayne P Burleson, and Kevin Fu. Power-up sram state as an identifying fingerprint and source of true random numbers. *IEEE Transactions on Computers*, 58(9) :1198–1210, 2008. 103

[32] Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger. Densely connected convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4700–4708, 2017. 79

[33] Yaseein Soubhi Hussein, Mohamad Yusoff Alias, and Ayad Atiyah Abdulkafi. On performance analysis of ls and mmse for channel estimation in vlc systems. In *2016 IEEE 12th International Colloquium on Signal Processing & Its Applications (CSPA)*, pages 204–209. IEEE, 2016. 46

[34] Taewon Hwang, Chenyang Yang, Gang Wu, Shaoqian Li, and Geoffrey Ye Li. Ofdm and its wireless applications : A survey. *IEEE transactions on Vehicular Technology*, 58(4) :1673–1694, 2008. 45, 46

[35] Suman Jana, Sriram Nandha Premnath, Mike Clark, Sneha K Kasera, Neal Patwari, and Srikanth V Krishnamurthy. On the effectiveness of secret key extraction from wireless signal strength in real environments. In *Proceedings of the 15th annual international conference on Mobile computing and networking*, pages 321–332. ACM, 2009. 87

[36] Gunes Z Karabulut and Abbas Yongacoglu. Sparse channel estimation using orthogonal matching pursuit algorithm. In *IEEE 60th Vehicular Technology Conference, 2004. VTC2004-Fall. 2004*, volume 6, pages 3880–3884. IEEE, 2004. 49

[37] Mohammad Khodaei and Panos Papadimitratos. The key to intelligent transportation : Identity and credential management in vehicular communication systems. *IEEE Vehicular Technology Magazine*, 10(4) :63–69, 2015. 85

[38] Beomjun Kim, Dongwook Kim, Sungyoul Park, Yonghwan Jung, and Kyongsu Yi. Automated complex urban driving based on enhanced environment representation with gps/map, radar, lidar and vision. *IFAC-PapersOnLine*, 49(11) :190–195, 2016. 60

[39] Dongsik Kim, Hyun-Myung Kim, and Gi-Hong Im. Iterative channel estimation with frequency replacement for sc-fdma systems. *IEEE transactions on communications*, 60(7) :1877–1888, 2012. 49

[40] Diederik P Kingma and Jimmy Ba. Adam : A method for stochastic optimization. *arXiv preprint arXiv :1412.6980*, 2014. 69

[41] Jan Kukačka, Vladimir Golkov, and Daniel Cremers. Regularization for deep learning : A taxonomy. *arXiv preprint arXiv :1710.10686*, 2017. 67

[42] Chan-Tong Lam, David D Falconer, and Florence Danilo-Lemoine. Iterative frequency domain channel estimation for dft-precoded ofdm systems using in-band pilots. *IEEE Journal on Selected Areas in Communications*, 26(2) :348–358, 2008. 49

[43] Yao-Nan Lee, Alexei Ashikhmin, and Jiunn-Tsair Chen. Impact of soft channel construction on iterative channel estimation and data decoding for multicarrier systems. *IEEE Transactions on Wireless Communications*, 7(7) :2762–2770, 2008. 48, 49

[44] Yanpei Liu, Stark C Draper, and Akbar M Sayeed. A secret key generation system based on multipath channel randomness : Rssi vs cssi. Technical report, 2011. 86

[45] Yinsheng Liu, Zhenhui Tan, Hongjie Hu, Leonard J Cimini, and Geoffrey Ye Li. Channel estimation for ofdm, 2014. 46, 48

[46] Ben Lu, Xiaodong Wang, and Ye Li. Iterative receivers for space-time block-coded ofdm systems in dispersive fading channels. *IEEE Transactions on Wireless Communications*, 1(2) :213–225, 2002. 49

[47] Xiongfeng Ma, Xiao Yuan, Zhu Cao, Bing Qi, and Zhen Zhang. Quantum random number generation. *npj Quantum Information*, 2(1) :1–9, 2016. 86

[48] Ueli Maurer and Stefan Wolf. Secret-key agreement over unauthenticated public channels-part iii : Privacy amplification. *IEEE Transactions on Information Theory*, 49(4) :839–851, 2003. 87

[49] du Développement durable et de l'Énergie Ministère de l'Écologie. Scoop, General presentation. http://www.scoop.developpement-durable.gouv.fr, 2016. 19

[50] Tom M Mitchell. Artificial neural networks. *Machine learning*, 45 :81–127, 1997. 61

[51] Hyung G Myung and David J Goodman. *Single carrier FDMA : a new air interface for long term evolution*, volume 8. John Wiley & Sons, 2008. 27

[52] Gaurang Naik, Biplav Choudhury, and Jung-Min Park. Ieee 802.11 bd & 5g nr v2x : Evolution of radio access technologies for v2x communications. *IEEE Access*, 7 :70169–70184, 2019. 26, 34, 35

[53] Magdalena Nohrborg. Lte. www.3gpp.org. 27

[54] Jorg Nuckelt, Moritz Schack, and Thomas Kurner. Performance evaluation of wiener filter designs for channel estimation in vehicular environments. In *2011 IEEE Vehicular Technology Conference (VTC Fall)*, pages 1–5. IEEE, 2011. 40

[55] Kazuhide Okamoto, Laurent Itti, and Panagiotis Tsiotras. Vision-based autonomous path following using a human driver control model with reliable input-feature value estimation. *IEEE Transactions on Intelligent Vehicles*, 4(3) :497–506, 2019. 60

[56] Mehmet Kemal Ozdemir and Huseyin Arslan. Channel estimation for wireless ofdm systems. *IEEE Communications Surveys & Tutorials*, 9(2) :18–48, 2007. 45

[57] Sriram Nandha Premnath, Suman Jana, Jessica Croft, Prarthana Lakshmane Gowda, Mike Clark, Sneha Kumar Kasera, Neal Patwari, and Srikanth V Krishnamurthy. Secret key extraction from wireless signal strength in real environments. *IEEE Transactions on mobile Computing*, 12(5) :917–930, 2012. 85, 87

[58] Soheyb Ribouh, Kelvin Phan, Arnav Vaibhav Malawade, Yassin El Hillali, Atika Rivenq, and Mohammad Abdullah Al Faruque. Channel State Information Based Cryptographic KeyGeneration for Intelligent Transportation Systems. *IEEE Transactions on Intelligent Transportation Systems*, 2020. to appear. 60

[59] Richard H Roy and Thomas Kailath. Esprit–estimation of signal parameters via rotational invariance techniques. *Optical Engineering*, 29(4) :296–314, 1990. 47, 48

[60] Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. Mobilenetv2 : Inverted residuals and linear bottlenecks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4510–4520, 2018. 79

[61] Frieder Sanzi, Sven Jelting, and Joachim Speidel. A comparative study of iterative channel estimators for mobile ofdm systems. *IEEE Transactions on Wireless Communications*, 2(5) :849–859, 2003. 48

[62] Axel Sauer, Nikolay Savinov, and Andreas Geiger. Conditional affordance learning for driving in urban environments. *arXiv preprint arXiv :1806.06498*, 2018. 60

[63] Ralph Schmidt. Multiple emitter location and signal parameter estimation. *IEEE transactions on antennas and propagation*, 34(3) :276–280, 1986. 48

[64] Torsten Schütze. Automotive Security : Cryptography for Car2x Communication. page 16. 98

[65] Young-Woo Seo, Jongho Lee, Wende Zhang, and David Wettergreen. Recognition of highway workzones for reliable autonomous driving. *IEEE Transactions on Intelligent Transportation Systems*, 16(2) :708–718, 2014. 60

[66] Barry Sheehan, Finbarr Murphy, Martin Mullins, and Cian Ryan. Connected and autonomous vehicles : A cyber-risk classification framework. *Transportation Research Part A : Policy and Practice*, 124 :523–536, June 2019. 84

[67] Youssef El Hajj Shehadeh, Omar Alfandi, Kifah Tout, and Dieter Hogrefe. Intelligent mechanisms for key generation from multipath wireless channels. In *2011 Wireless Telecommunications Symposium (WTS)*, pages 1–6. IEEE, 2011. 86

[68] Youssef El Hajj Shehadeh and Dieter Hogrefe. A survey on secret key generation mechanisms on the physical layer in wireless networks. *Security and Communication Networks*, 8(2) :332–341, 2015. 87

[69] Marcos A Simplicio, Eduardo Lopes Cominetti, Harsh Kupwade Patil, Jefferson E Ricardini, and Marcos Vinicius M Silva. The unified butterfly effect : Efficient security credential management system for vehicular communications. In *2018 IEEE Vehicular Networking Conference (VNC)*, pages 1–8. IEEE, 2018. 85

[70] HZ Bo Sun. Ieee 802.11-18/0861r9 : 802.11 ngv proposed par. In *IEEE 802.11 NGV Meeting*, 2019. 26

[71] Qinfang Sun, Donald C Cox, Howard C Huang, and Angel Lozano. Estimation of continuous flat fading mimo channels. In *2002 IEEE Wireless Communications and Networking Conference Record. WCNC 2002 (Cat. No. 02TH8609)*, volume 1, pages 189–193. IEEE, 2002. 46

[72] Manisha B Sutar and Vikram S Patil. Ls and mmse estimation with different fading channels for ofdm system. In *2017 International conference of Electronics, Communication and Aerospace Technology (ICECA)*, volume 1, pages 740–745. IEEE, 2017. 46

[73] Christian Szegedy, Sergey Ioffe, Vincent Vanhoucke, and Alex Alemi. Inception-v4, inception-resnet and the impact of residual connections on learning. *arXiv preprint arXiv :1602.07261*, 2016. 79

[74] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2818–2826, 2016. 79

[75] Stephan Ten Brink. Convergence of iterative decoding. *Electronics letters*, 35(10) :806–808, 1999. 49

[76] M Tüchler, S Ten Brink, and J Hagenauer. Measures for tracing convergence of iterative decoding algorithms. In *in Proc. 4th IEEE/ITG Conf. on Source and Channel Coding*. Citeseer, 2002. 49

[77] Meltem Sönmez Turan. Recommendation for the entropy sources used for random bit generation. *NIST Special Publication*, 800 :90B, 2018. 102

[78] Robert Varga, Arthur Costea, Horatiu Florea, Ion Giosan, and Sergiu Nedevschi. Super-sensor for 360-degree environment perception : Point cloud segmentation using image features. In *2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSC)*, pages 1–8. IEEE, 2017. 60

[79] Jon W Wallace, Chan Chen, and Michael A Jensen. Key generation exploiting mimo channel evolution : Algorithms and theoretical limits. In *2009 3rd European Conference on Antennas and Propagation*, pages 1499–1503. IEEE, 2009. 86

[80] Jon W Wallace and Rajesh K Sharma. Automatic secret keys from reciprocal mimo wireless channels : Measurement and analysis. *IEEE Transactions on Information Forensics and Security*, 5(3) :381–392, 2010. 86

[81] Jiang Wan, Anthony Lopez, and Mohammad Abdullah Al Faruque. Physical layer key generation : Securing wireless communication in automotive cyber-physical systems. *ACM Transactions on Cyber-Physical Systems*, 3(2) :1–26, 2018. 85

[82] Jiang Wan, Anthony Bahadir Lopez, and Mohammad Abdullah Al Faruque. Exploiting wireless channel randomness to generate keys for automotive cyber-physical system security. In *2016 ACM/IEEE 7th International Conference on Cyber-Physical Systems (ICCPS)*, pages 1–10. IEEE, 2016. 44, 88, 92, 93

[83] R Wolke and H Schwetlick. Iteratively reweighted least squares : algorithms, convergence analysis, and numerical comparisons. *SIAM journal on scientific and statistical computing*, 9(5) :907–921, 1988. 49

[84] Xiaohua Wu, Yuexing Peng, Chunjing Hu, Hui Zhao, and Lei Shu. A secret key generation method based on csi in ofdm-fdd system. In *2013 IEEE Globecom Workshops (GC Wkshps)*, pages 1297–1302. IEEE, 2013. 85

[85] Imam Uz Zaman, Anthony Bahadir Lopez, Mohammad Abdullah Al Faruque, and Ozdal Boyraz. A physical layer security key generation technique for inter-vehicular visible light communication. In *Signal Processing in Photonic Communications*, pages SpTu1F–3. Optical Society of America, 2017. 85

[86] Imam Uz Zaman, Anthony Bahadir Lopez, Mohammad Abdullah Al Faruque, and Ozdal Boyraz. Polarization mode dispersion-based physical layer key generation for optical fiber link security. In *Optical Sensors*, pages JTu4A–20. Optical Society of America, 2017. 85

[87] Imam Uz Zaman, Anthony Bahadir Lopez, Mohammad Abdullah Al Faruque, and Ozdal Boyraz. Physical layer cryptographic key generation by exploiting pmd of an optical fiber link. *Journal of Lightwave Technology*, 36(24) :5903–5911, 2018. 85

[88] Thomas Zemen, Laura Bernadó, Nicolai Czink, and Andreas F Molisch. Iterative time-variant channel estimation for 802.11 p using generalized discrete prolate spheroidal sequences. *IEEE Transactions on Vehicular Technology*, 61(3) :1222–1233, 2012. 40

[89] Yi Zhang, Ping Sun, Yuhan Yin, Lin Lin, and Xuesong Wang. Human-like autonomous vehicle speed control by deep reinforcement learning with double q-learning. In *2018 IEEE Intelligent Vehicles Symposium (IV)*, pages 1251–1256. IEEE, 2018. 60

[90] Jizhong Zhao, Wei Xi, Jinsong Han, Shaojie Tang, Xiangyang Li, Yunhao Liu, Yihong Gong, and Zehua Zhou. Efficient and secure key extraction using csi without chasing down errors. *arXiv preprint arXiv :1208.0688*, 2012. 86

[91] Hao Zhu, Ka-Veng Yuen, Lyudmila Mihaylova, and Henry Leung. Overview of environment perception for intelligent vehicles. *IEEE Transactions on Intelligent Transportation Systems*, 18(10) :2584–2601, 2017. 60

[92] Xiangyang Zhuang and Frederick W Vook. Iterative channel estimation and decoding for a turbo-coded ofdm system via the em algorithm. In *2002 IEEE International Conference on Acoustics, Speech, and Signal Processing*, volume 3, pages III–2337. IEEE, 2002. 49

[93] Yanying Zhuang, Jingyu Hua, Hao Wen, Limin Meng, et al. An iterative doppler shift estimation in vehicular communication systems. *Procedia Engineering*, 29 :4129–4134, 2012. 40