



REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique
Université Mohamed Khider – BISKRA
Faculté des Sciences Exactes, des Sciences de la Nature et de la Vie
Département d'informatique

N° d'ordre : /M2/2020

Mémoire

Présenté pour obtenir le diplôme de master académique en

Informatique

Parcours : RTIC

La navigation des véhicules autonomes par un algorithme génétique

Par :

GUERFI NASSIRA

Soutenu le _____, devant le jury composé de :

BITAM Salim

Professeur

Président

Rapporteur

Examineur

Table des matières

1	L'Internet des véhicules, les véhicules autonomes et la communication V2X	11
1.1	Introduction	11
1.2	Définition de l'IoV	12
1.3	Caractéristiques de l'IoV	13
1.4	Architecture en couches et pile de protocoles	14
1.4.1	Architecture en couches	14
1.4.2	Pile de protocoles	16
1.5	Modèle de réseau de l'IoV	19
1.6	Les modes de communications dans l'IOV	20
1.7	Le véhicule autonome	22
1.7.1	Définition	22
1.7.2	Classification des niveaux d'automatisation	22
1.7.3	Problématiques des véhicules autonomes	23
1.8	La navigation des véhicules autonomes	24
1.8.1	Définition	24
1.8.2	La navigation globale et la perception locale	25
1.8.3	La navigation des véhicules autonomes basée sur la communication V2X	30
1.9	Conclusion	30
2	L'état de l'art	31

2.1	Introduction	31
2.2	Les entités de communications	31
2.2.1	unité à bord OBU(On Board Unit)	33
2.2.2	unité d'application AU (Application Unit)	34
2.2.3	une unité côté route RSU(Road Side Unit)	34
2.3	Les approches utilisées pour la communication V2X	36
2.3.1	La navigation des véhicules autonomes basée sur la communication V2V .	37
2.3.2	La navigation des véhicules autonomes basée sur la communication V2P .	38
2.3.3	La navigation des véhicules autonomes basée sur la communication V2I .	39
2.4	Conclusion	41
3	Conception et développement	42
3.1	Introduction	42
3.2	La conception générale du système	42
3.3	La conception détaillée du système	43
3.3.1	Modélisation de l'environnement	44
3.3.2	Algorithme de djikstra pour la navigation	45
3.3.3	Algorithme génétique pour contourner un obstacle	46
3.3.4	Communication V2X du chemin découvert	49
3.4	Conclusion	50
4	Implémentation et résultats	51
4.1	Introduction	51
4.2	Environnement d'expérimentation	51
4.2.1	Environnement matériel	52
4.2.2	Environnement logiciel	52
4.3	La navigation communicante de véhicule autonome par un algorithme génétique :	54
4.3.1	Modélisation de l'environnement de navigation :	54

4.3.2	Développement et implémentation de la navigation globale :	54
4.3.3	Développement et implémentation de la navigation locale :	57
4.3.4	la communication V2X du chemin découvert	59
4.4	Expérimentation et résultats	59
4.4.1	Scénarios d'une expérimentation	59
4.4.2	Résultats obtenus	61
4.5	Conclusion	63

Table des figures

1.1	Un réseau IoV	12
1.2	Caractères de IIOV	13
1.3	L'architecture en cinq couches de l'IoV	14
1.4	La pile de protocole d'IoV	17
1.5	Les trois éléments de réseau de l'IoV	19
1.6	Le modèle de réseau d'IoV	20
1.7	Les cinq types de communication de l'IoV	20
1.8	La réalisation de l'IoV avec des réseaux véhiculaires hétérogènes [4]	21
1.9	Les niveaux d'automatisation selon SAE international	23
1.10	schéma générale de la navigation autonome	25
1.11	navigation globale (délibérative)[8]	26
1.12	modélisation de l'environnement de la navigation de véhicule autonome	26
1.13	les représentations de l'environnement : la représentation topologique mémorise des lieux et des liens de déplacement, dans cet exemple B=Bureau, P=Porte, C=Couloir, I=Intersection.les représentation métrique mémorise des objets (murs) avec une position dans un repère global	28
2.1	Les entités de communications	32
2.2	On Board Unit(OBU)	33
2.3	Road Side Unit (RSU)	34
2.4	RSU étend la portée du réseau ad-hoc en transmettant les données entre les OBUs	35

2.5	fonctionne comme source d'information(exécution d'application de sécurité)	35
2.6	RSU fournit une connectivité Internet aux OBUs	36
2.7	trois modes de la communication V2X[12]	37
2.8	mode de communication V2V	38
2.9	le mode de communication V2I	40
3.1	Conception générale du système	43
3.2	Conception détaillée du système	44
3.3	Fonctionnement de l'algorithme génétique	47
3.4	Représentation de la Croisement	48
3.5	Représentation de la mutation	49
3.6	modèle de la communication V2X	50
4.1	Flot de simulation avec NS-2	53
4.2	Modèle de mobilité du véhicule	60
4.3	L'itinéraire le plus court entre le point de départ et la destination trouvé par l'algorithme de Djikstra	61
4.4	L'itinéraire le plus court entre le point de départ et la destination trouvé par l'algorithme génétique	62

Dédicaces

Je dédie ce travail A mes très chers parents que je ne peux jamais autant

Remercier

A mes frères

A mes Soeurs

A tous mes amis

A tous mes enseignants

Remerciements

En premier lieu, je remercie le Bon Dieu ALLAH de m'avoir donné la force et le courage pour accomplir ce travail et qui m'a procuré ce succès.

Je tiens à exprimer mon reconnaissance envers mon encadreur le Professeur Salim Bitam pour avoir encadré, suivi et dirigé pendant toute la durée de ce travail.

Je tiens également à remercier tous nos enseignants, ainsi que tous les membres du département de l'informatique.

Je tiens à remercier l'ensemble des Membres du Jury pour m'avoir fait l'honneur de leur présence et pour l'intérêt qu'ils ont porté à mon mémoire en acceptant d'examiner ce travail, et de l'enrichir par leurs propositions.

J'adresse bien évidemment mes remerciements à mes parents, mes frères, mes soeurs, mes familles, aussi mes amis qui ont tous su me soutenir sans insister et s'informer de l'état d'avancement de mes travaux sans être trop curieux.

Je ne pourrais cloturer ces remerciements sans me retourner vers les êtres qui me sont le plus chers, qui ont eu un rôle essentiel et continu pendant ma réussite, et qui sans eux aucune réussite n'aurait été possible. J'adresse de tout mon coeur mes remerciements à mes chers parents, je suis infiniment reconnaissant pour leurs amours, leur soutiens moraux, qu'ils trouvent dans ce travail le fruit de leur travail.

Résumé

La navigation des véhicules autonomes basée sur la communication V2x est un axe de recherche qui vise à donner la possibilité à un véhicule de se déplacer dans un environnement routier sans l'intervention de l'être humain et par l'aide de la communication de données distante.

Cette mémoire de Master s'intéresse à la planification de trajectoire pour assurer une navigation autonome d'un véhicule basée sur la communication V2x. Ce système commence par un calcul de plus court chemin en appliquant l'algorithme de Dijkstra et puis l'algorithme génétique en utilisant les informations statiques de l'environnement et la communication V2X entre les noeuds du réseau.

Le véhicule peut échanger les données du chemin trouvé et transmettre ses informations (débit, délai, etc.) vers tous les noeuds de l'environnement routière grâce aux différentes interfaces réseaux telles que le WAVE/DSRC(Wireless Access in VehicularEnvironments/Dedicated Short Range Communication), WIFI, 4/5G au via les différents points d'accès des RSU déployées aux bords des routes.

Les résultats obtenus comparés à l'algorithme de navigation traditionnel Dijkstra ont montré l'efficacité de notre proposition.

Introduction générale

Les système de transport intelligents (STI) constitue un domaine des nouvelles technologies de l'information et de la communication appliquées au transport dans le but d'améliorer la sécurité routière et pour servir les usagers de la routes par la communication véhicule to autres noeuds dans le réseau (connue sous le nom communication V2X). Parmi les avancées récentes dans ce domaine, on cite les véhicules autonomes qui ont pour but la navigation sans aucune intervention humaine.

Dans ce mémoire , nous allons étudier comment Le véhicule échanger les données du l'environnement avec l'utilisation de la communication V2X pour planifier le plus court chemin entre le point de départ et la destination.

A cet effet, nous proposons de développer et concevoir un système de navigation qui combine entre une méthode de planification globale basée sur l'algorithme de Dijkstra et une méthode locale basée sur l'algorithme génétique afin de trouver un itinéraire alternatif en utilise la communication V2X.

Notre mémoire est organisé en quatre chapitres :

Dans Le premier chapitre, nous consacrons à la présentation du concept de l'Internet des véhicules, les véhicules autonome et la communication V2X.

Dans le deuxième chapitre, nous présente les entités de communication et quelques approches utilisées pour la communication V2X.

Dans le troisième chapitre, Nous décrivons l'architecture générale et détaillée du système, la méthode proposée pour résoudre le problème de la navigation des véhicules autonomes basée sur

la communication V2X.

Dans le chapitre quatre, nous exposerons la tâche finale de ce projet, qui est la réalisation des objectifs suivants de notre système, en présentant l'environnement pour le développement et l'implémentation de notre système avec les différents résultats obtenus.

On terminera Ce travail par une conclusion générale.

Chapitre 1

L'Internet des véhicules, les véhicules autonomes et la communication V2X

1.1 Introduction

La technologie de l'Internet des véhicules (IoV) a évolué au fil du temps à partir du réseau classique ad hoc de véhicules (VANET), qui fait référence au réseau de différentes entités de transport routier, telles que les véhicules, les stations de base installées au bord de la route (Road side units -RSU-), les piétons, etc. L'objectif principal de l'IoV est d'assurer la sécurité routière et le confort des usagers de la route par une communication appelée communication véhicule à toute entité (V2X) qui doit être fiable et en temps réel. La communication V2X est utilisée pour permettre la communication entre les véhicules, les infrastructures, les capteurs, c'est-à-dire tout ce qui entoure le véhicule.

Dans ce chapitre, nous allons commencer par l'illustration des concepts de base de l'IoV et leurs caractères, puis, nous expliquons l'architectures en couches et la pile de protocoles, le modèle de réseau et les modes de communication dans l'IoV, enfin, nous étudions la navigation des véhicules autonomes basée sur la communication V2X comme étant une application performante de l'IoV.

1.2 Définition de l'IoV

L'Internet de Véhicule (IoV) est un système cyber-physique émergent qui intègre les réseaux VANET, l'IoT et l'informatique en nuage mobile. Dans l'IoV, chaque véhicule est considéré comme un objet intelligent avec une puissante plate-forme multi-capteurs, de technologies des communications, d'unités de calcul, de connectivité IP ternet et à d'autres véhicules, directement ou indirectement. La figure 1.1 montre un réseau IoV [1] :

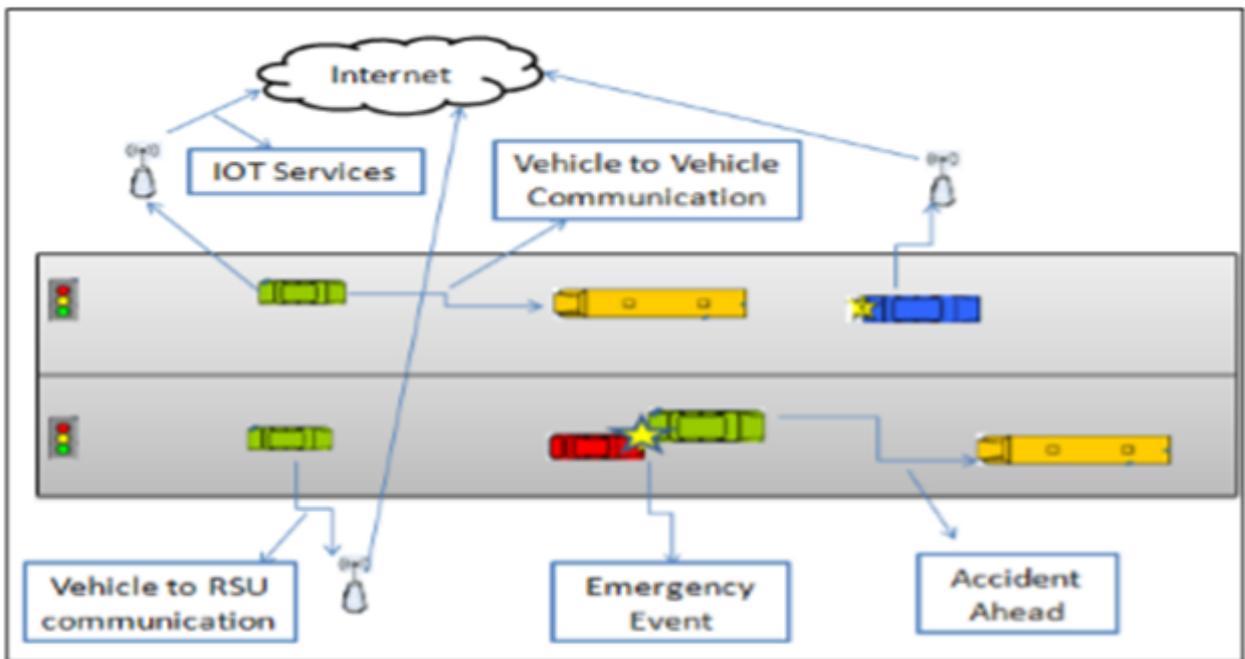


FIGURE 1.1 – Un réseau IoV

L'IoV intègre trois réseaux : un réseau inter-véhicules, un réseau intra-véhicules et un réseau Internet mobile. Sur la base de ce concept de trois réseaux intégrés en un seul, l'IoV est défini comme un système distribué à grande échelle pour la communication sans fil et l'échange d'informations entre les véhicules et les autres noeuds du réseau (V2X) (X : véhicule, station de base routière, personne, capteur, infrastructure) conformément aux protocoles de communication et aux normes de données convenus[2].

1.3 Caractéristiques de l’IoV

Les caractéristiques spécifiques de l’IoV sont (voir figure 1.2) [3] :

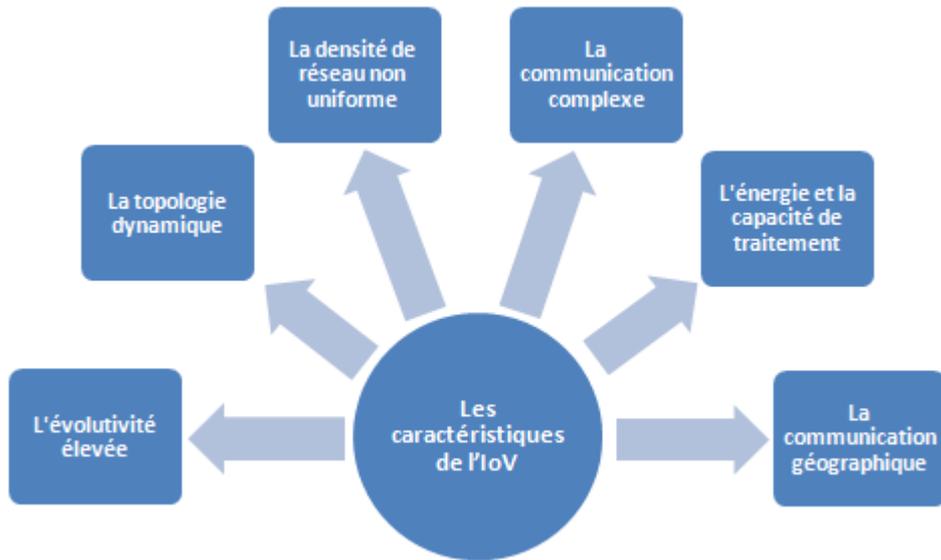


FIGURE 1.2 – Caractères de IIoV

- Évolutivité élevée : dans une grande ville, il peut avoir des millions des véhicules, et pour créer un réseau de véhicules, il est nécessaire d’avoir un réseau à grande échelle et évolutif.
- Topologie dynamique : de nombreux composants hétérogènes différents d’IoV interagissent les uns avec les autres et ces composants (en particulier les véhicules) se déplacent à grande vitesse, ce qui modifie rapidement la topologie du réseau.
- Densité de réseau non uniforme : La densité de réseau IoV varie en fonction de diverses conditions telles que la grandeur de la ville, les autoroutes, la localisation géographique, etc. Cette non-uniformité doit être prise en compte lors de la conception de tout réseau de véhicules IoV.
- Communication complexe : La densité du réseau IoV varie d’un scénario à l’autre. L’emplacement des véhicules et la distance respective varient à grande vitesse, ce qui nécessite une vitesse de communication élevé avec un retard minimal. L’IoV est alors un réseau de communication très complexe qui doit être extrêmement fiable.

- Énergie et capacité de traitement : le réseau de véhicules IoV ne manque pas d'énergie, de puissance de traitement ou de capacité de mémoire. Les noeuds sont des véhicules ou des plates-formes qui disposent de suffisamment d'énergie et d'espace pour inclure une puissance de traitement et une mémoire importante.
- Communication géographique : le réseau véhiculaire utilise la communication géographique dans laquelle les paquets sont transférés d'un noeud à un autre. Dans le cas d'IoV, les noeuds ne sont pas prédéfinis et varient rapidement en fonction des emplacements géographiques.

1.4 Architecture en couches et pile de protocoles

1.4.1 Architecture en couches

L'architecture de l'IoV contient 5 niveaux : la couche de perception, la couche de coordination, la couche d'intelligence artificielle (IA), la couche d'application et la couche des affaires. Les représentations (figure 1.3) et les fonctionnalités de chaque couche sont détaillé ci-dessous [4].

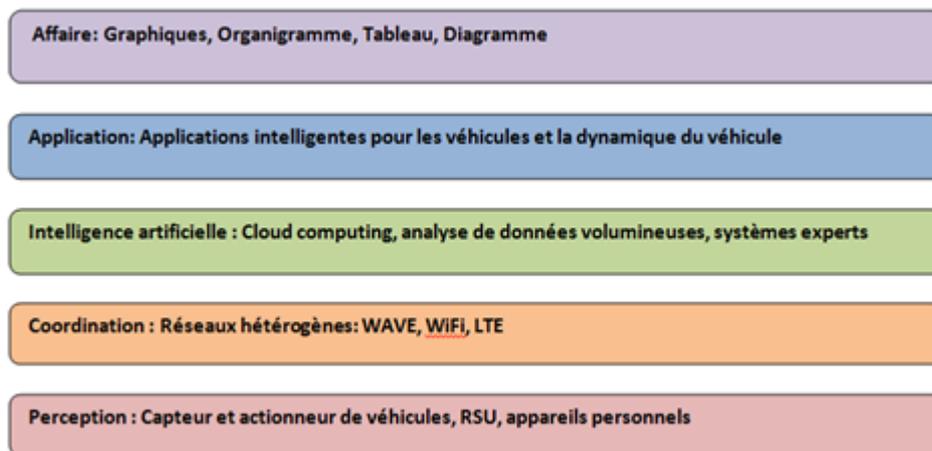


FIGURE 1.3 – L'architecture en cinq couches de l'IoV

1.4.1.1 Couche de perception

La première couche de l'architecture de l'IoV est représentée par les capteurs et les actionneurs attachés aux véhicules, RSU, Smartphones et autres appareils personnels qui se présentent dans

l'environnement routier.

- Rôle
 - rassembler des informations sur le véhicule, l'environnement de la circulation y compris les données liées au mouvement.

1.4.1.2 Couche de coordination

La deuxième couche de l'architecture est représentée par un module de coordination de réseaux universels virtuels qui sont souvent hétérogènes tels que les réseaux WAVE, Wi-Fi, 4G / LTE et satellites.

- Rôle
 - Assurer la tâche de transport de données
 - Assurer la tâche de traitement d'informations reçues dans un réseau hétérogènes (het-nets) qui nécessite de créer une structure unifiée avec des capacités d'identification pour chaque type de réseau
 - Il est à noter qu'il y a un manque de normes pour cette couche dont le but est d'assurer l'interopérabilité et la coopération entre les différents types de réseaux pour avoir une connectivité réseau fiable

1.4.1.3 Couche intelligence artificielle (AI)

La troisième couche de l'architecture est représentée par l'infrastructure virtuelle du cloud.

- Rôle
 - stocker, traiter et analyser les informations reçues de la couche inférieure
 - assurer la prise de décision au profit des utilisateurs des l'IoV
 - Cela fonctionne comme centre de gestion de l'information ; les principaux composants sont : Cloud Computing véhiculaire (VCC), analyse de données volumineuses (BDA), système expert

1.4.1.4 Couche d'application

La quatrième couche de l'architecture de l'IoV est représentée par des applications intelligentes, assurant la sécurité du trafic, l'efficacité, et des services multimédia comme les applications utilitaires basées sur le Web.

- Rôle
 - fournir des services intelligents aux utilisateurs sur la base d'une analyse intelligente effectuée par l'IA
 - fournir des applications de sécurité routière

1.4.1.5 Couche d'affaires

La cinquième couche est la couche des affaires, elle est représentée par un module de gestion opérationnelle de l'IoV.

- Rôle
 - prévoir des stratégies de développement de modèles commerciaux basés sur les données
 - offrir des outils d'analyse, notamment des graphiques, des organigrammes, des tableaux de comparaison, des diagrammes de cas d'utilisation, etc.
 - aider à la prise de décision en matière d'investissement économique et d'utilisation des ressources
 - aider à la tarification, préparation du budget global pour l'exploitation et la gestion de données

1.4.2 Pile de protocoles

Une pile de protocoles est conçue en organisant de manière efficace les protocoles existants appropriés dans l'une des cinq couches précédentes (voir la Figure 1.4). La pile pour l'architecture en couches de l'IoV est structuré en considérant des plans de gestion, opérationnels et de sécurité [4].

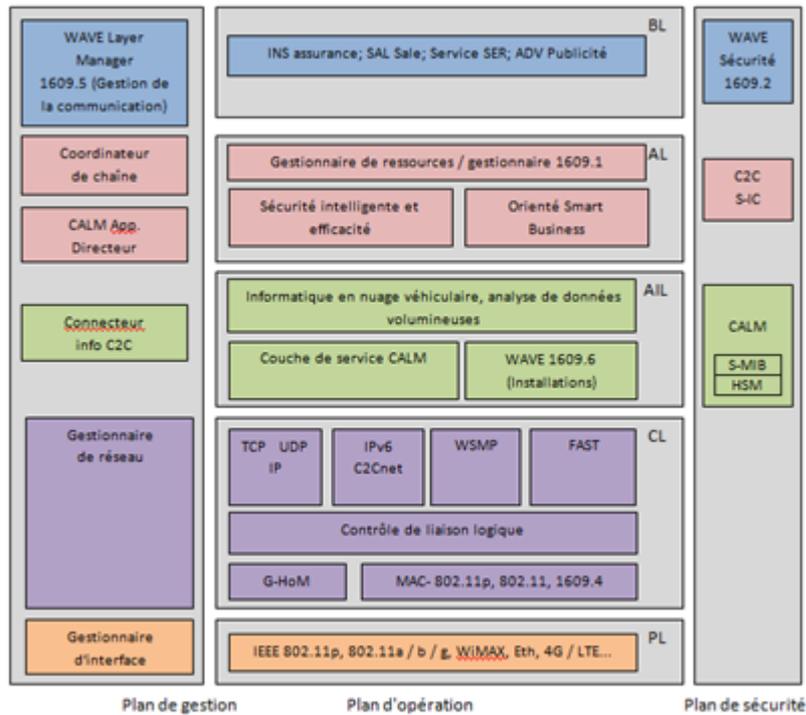


FIGURE 1.4 – La pile de protocole d’IoV

1.4.2.1 Plan d’opération

- Couche de perception : un certain nombre de protocoles d’accès sans fil pourraient être pris en compte : Dans la couche physique, notamment 802.11p de WAVE, 802.11a/b/g de WLAN, WiMax, Ethernet, 4G/ LTE et GSM
- Couche de coordination : est divisé en trois sous-couches : inférieure, moyenne et supérieure. Les différents protocoles MAC pourraient être utilisés à la sous-couche inférieure. Un protocole de contrôle de liaison logique (LLC) pourrait être envisagé au niveau de la sous-couche intermédiaire (moyenne). Aux sous-couches supérieures, trois protocoles, dont le protocole de réseau C2C (C2C-net), le protocole de messages courts (WSMP) et le facilitateur de communication (FAST), pourraient être envisagés pour réaliser les opérations des couches réseau et transport.
- Couche d’intelligence artificielle : les protocoles liés au service CALM-SL (CALM Service Layer) et WAVE-1609.6 seraient pris en compte dans la sous-couche inférieure, ainsi que des

protocoles liés au Cloud Computing, au Vehicular Cloud Computing (VCC) et au Big Data Analysis (BDA) pourraient être utilisés au niveau de la sous couche supérieure, y compris les services en nuage Stockage en tant que service (STaaS), infrastructure en tant que service (INaaS), réseau en tant que service (NaaS)

- Couche d'application : utilisé un protocole défini dans WAVE qui pourrait être utilisé pour gérer les ressources entre les applications intelligentes.
- Couche de gestion : quatre types de modèles d'entreprises sont prises en compte ce qui constituent un défi de recherche ouvert en IoV, ces différents modèles peuvent être définis comme : l'assurance (INS), la vente (SAL) , le service (SER) et la publicité (ADV).Le succès d'IoV dépend fortement du développement efficace de ces modèles commerciaux qui sont les plus importants pour la commercialisation de l'IoV dans les industries connexes.

1.4.2.2 Plan de sécurité

Ce plan est un défi de recherche ouvert (indisponibilité de définitions claires de protocoles de sécurité entre couches). Actuellement, les protocoles de sécurité (provenant de WAVE, C2C et CALM) peuvent être utilisés tels que le IEEE 1609.2, Sécurité-connecteur d'Information (S-IC), Sécurité-gestion de base d'informations (S-MIB) et matériel de sécurité sécurisé (HSM).

1.4.2.3 Plan de gestion

Dans ce plan, trois groupes de protocoles appartenant à des opérations de gestion dont WAVE, CALM et C2C pourraient être considérés. Les protocoles de gestion de couche IEEE 1609.5 et de coordinateur de canal sont développés dans WAVE. CALM a suggéré trois protocoles : les gestionnaires d'applications, de réseaux et d'interfaces. Le protocole de connecteur d'information a été suggéré dans C2C dans le même but.

1.5 Modèle de réseau de l’IoV

Un modèle de réseau de l’IoV est proposé en identifiant Les trois principaux éléments de réseau d’IoV : le cloud computing (l’informatique en nuage), la connexion et le client comme le montre dans la figure 1.5 [4]. Ces éléments et leurs rôles dans IoV sont décrits en détail ci-dessus.

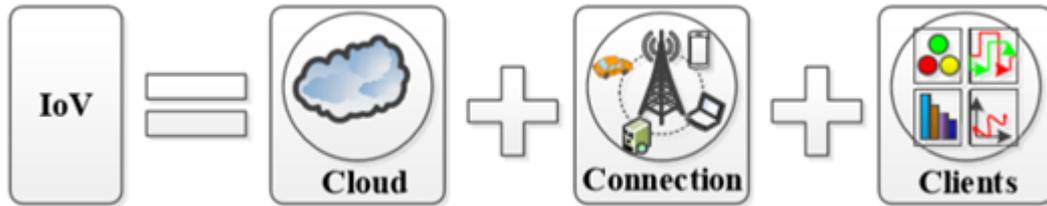


FIGURE 1.5 – Les trois éléments de réseau de l’IoV

L’infrastructure cloud, qui est Le premier élément d’IoV fournit une plate-forme pour une gamme de technologies d’accès sans fil. étant donné que l’ampleur des informations liées au trafic est susceptible d’augmenter considérablement, il est idéal de gérer les informations à l’aide du cadre informatique en nuage. Par ailleurs, Le deuxième élément de l’IoV est la Connexion qui utilise l’opérateur de réseau tiers (TPNIO) pour réduire les accords de niveau de service directs entre les opérateurs des réseaux, permettant ainsi une itinérance continue sans compromettre la qualité et la sécurité des opérateurs de réseau. L’élément client qui constitue le troisième élément de l’IoV, Chaque application client a un service exigences qui peuvent être différentes des autres clients. Les exigences de service d’un client sont définies en termes des caractéristiques d’une technologie d’accès sans fil. Par conséquent, une préférence prioritaire des technologies d’accès sans fil est utilisée par les applications clientes.

Une vue logique du modèle de réseau proposé d’IoV est illustrée à la figure 1.6 avec les composants internes de chaque élément [4].

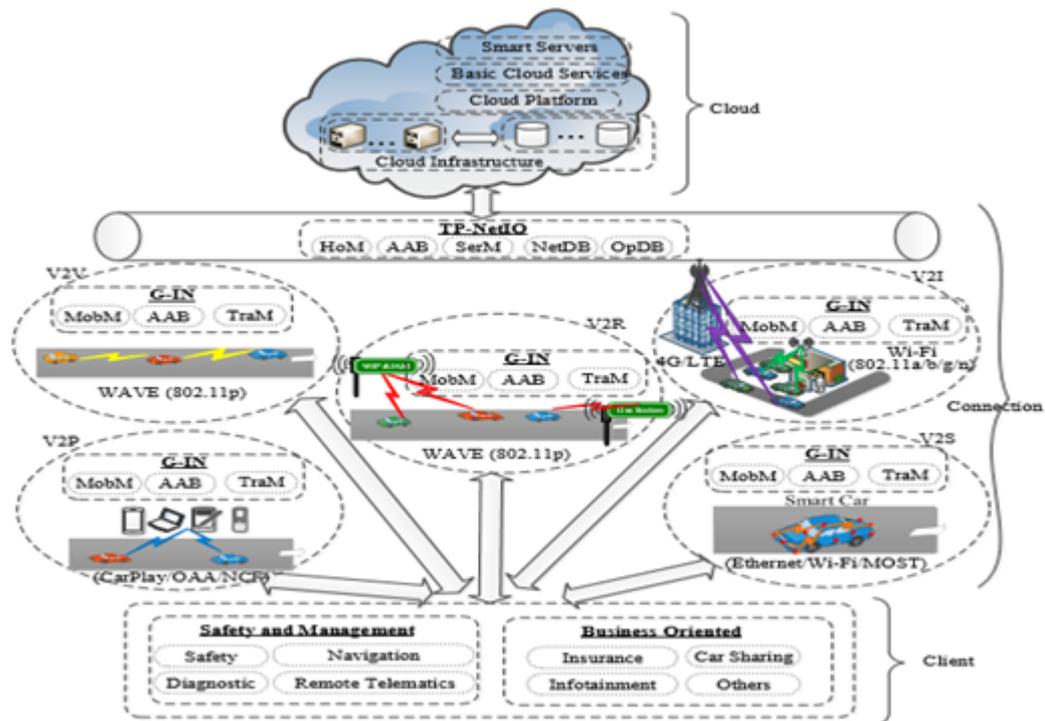


FIGURE 1.6 – Le modèle de réseau d'IOV

1.6 Les modes de communications dans l'IOV

L'architecture de réseau hétérogène d'IOV comprend cinq types de communications véhiculaires. Les types sont : véhicule à véhicule (V2V), véhicule à unité de route (V2R), véhicule à infrastructure (V2I) de réseaux mobiles, véhicule à piéton (V2P) et véhicule à capteur (V2S) (voir Figure 1.7) [4].

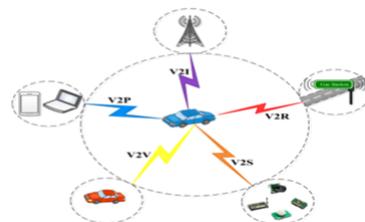


FIGURE 1.7 – Les cinq types de communication de l'IOV

Le réseau véhiculaire hétérogène de l'IOV présenté a un potentiel important pour guider et superviser les véhicules par sa capacité de fournir une plate-forme de communication fiable pour Internet mobile et applications liées au multimédia.

Chaque communication véhiculaire d'IoV est activée à l'aide d'une technologie WAT (Wireless Access Technologies) différente. WAT inclue les communications :

- IEEE WAVE pour V2V et V2R
- Wi-Fi et 4G / LTE pour V2I
- CarPlay / NCF pour V2P
- MOST / Wi-Fi pour V2S

L'architecture de communication n'inclut que des véhicules et des unités de route RSU, mais également d'autres dispositifs de communication. La figure 1.8 présente un exemple de ce cadre avec une coordination de réseau hétérogène et des services associés.



FIGURE 1.8 – La réalisation de l'IoV avec des réseaux véhiculaires hétérogènes [4]

1.7 Le véhicule autonome

1.7.1 Définition

Un véhicule autonome est un véhicule sans conducteur qui permet de se déplacer d'un point de départ à un point d'arrivé sans intervention humaine. Pour permettre de circuler dans des conditions réelles, le véhicule autonome est équipé d'un système de pilotage automatique. Celui-ci est alors composé d'un certain nombre de capteurs, caméras et radars lui servant à modéliser l'environnement global qui l'entoure et de différents systèmes de télécommunication pour échanger des informations à courtes et à longues distances.

Il existe deux types de véhicules autonomes [5] : le véhicule autonome pur (self-contained) et le véhicule autonome connecté. Le premier type s'appuie uniquement sur les informations générées par le véhicule lui-même, tandis que le deuxième type implique que le véhicule est connectée à un ou plusieurs réseaux de communication, le véhicule ret des informations sur l'infrastructure, les autres véhicules, pour l'aide à la conduite.

1.7.2 Classification des niveaux d'automatisation

Les avancées technologiques dans le secteur de l'automobile créent un continuum entre la voiture conventionnelle, c'est-à-dire la voiture qui nécessite le contrôle du conducteur en permanence et la voiture totalement autonome. La Society of Automotive Engineers (SAE)[6] comporte six niveaux qui vont de la conduite toute humaine à la conduite toute automatisée. Elle évalue le degré d'attention et d'intervention requis du conducteur plutôt que des fonctions intégrées au véhicule, bien que ces deux éléments soient étroitement liés.

Niveau 0 :

L'ensemble de la conduite est assurée par le conducteur

Niveau 1 :

Une fonction assure le contrôle de la direction du véhicule ou de l'accélération/décélération

Niveau 2 :

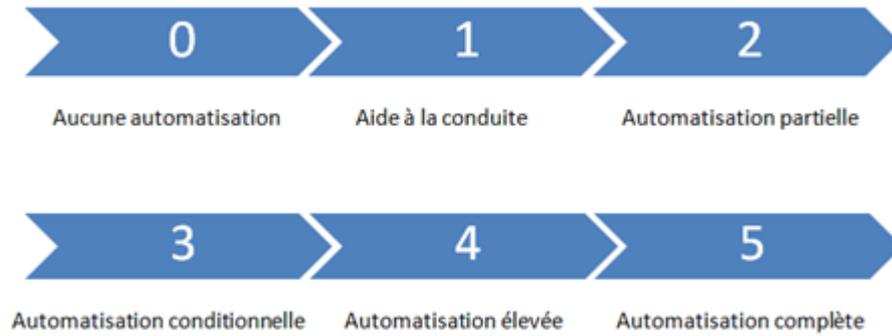


FIGURE 1.9 – Les niveaux d’automatisation selon SAE international

Une ou plusieurs fonctions assurent le contrôle de la direction du véhicule ou de l’accélération/décélération mais le conducteur exécute les tâches dynamiques de conduite

Niveau 3 :

Le système de conduite automatisé accomplit l’ensemble des tâches de la conduite selon les circonstances et le trafic routier, dans ce niveau et le conducteur doit être disponible pour intervenir en cas de besoin

Niveau 4 :

Le système de conduite automatisé accomplit l’ensemble des tâches de la conduite selon les circonstances et le trafic routier même si le conducteur n’est pas disponible pour intervenir et conduire au besoin

Niveau 5 :

Le système de conduite automatisé accomplit la totalité de la conduite dans toutes les circonstances sans la nécessité de l’intervention du conducteur.

1.7.3 Problématiques des véhicules autonomes

Les véhicules autonomes pourraient être commandés et venir chercher leurs passagers. Toutefois, bien que le véhicule autonome présente de nombreux atouts, cette technologie fait aussi apparaître quelques problématiques[7] :

- L’apparition de nombreux trajets d’ajustement et redondants (par exemple, les trajets pour

aller chercher quelqu'un qui a commandé une voiture, pour aller se garer, pour anticiper une demande dans un lieu éloigné)

- Une réticence possible des utilisateurs à laisser le contrôle de leur voiture à des machines
- Le piratage informatique des voitures, ciblé contre un véhicule, ou contre toute flotte simultanément
- Les voitures ne peuvent pas pour l'instant obéir aux signaux et injonctions des agents de police. Dans ce cas, ces derniers devront disposer d'outils adaptés pour réguler la circulation ou arrêter le véhicule
- Le surcoût d'un véhicule autonome par rapport à son équivalent non autonome pourrait être un frein auprès de la clientèle
- La requalification des conducteurs
- La transformation du secteur de la vente et de l'entretien des véhicules

1.8 La navigation des véhicules autonomes

1.8.1 Définition

La navigation autonome d'un véhicule requiert l'intation des trois grandes familles de processus de perception, de planification et d'action. Cette navigation est de permettre, à partir de la perception de l'environnement, de rechercher une trajectoire dans un espace libre entre un point initial et un point final, puis de se déplacer sur cette trajectoire. Si la perception se fait au niveau du véhicule, la planification de mouvement peut se prendre de fa partagée entre l'opérateur et le véhicule. L'autonomie peut se définir comme la capacité d'un système à réagir par lui-même à ses propres observations de l'environnement, sans intervention humaine.

Les trois actions majeures de la navigation autonome sont : la perception, la planification et l'action.

La perception

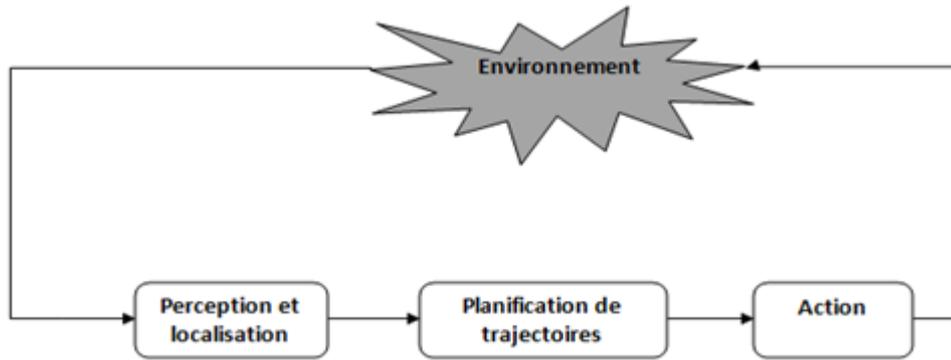


FIGURE 1.10 – schéma générale de la navigation autonome

permet de fournir une carte dynamique de l’environnement proche du véhicule. La localisation permet au véhicule de se repérer sur la base d’un repère global, ce qui permettrait de définir les itinéraires possibles de navigation pour atteindre la destination.

La planification de trajectoire

consiste à calculer une trajectoire de référence, qui évite les obstacles, cette étape permet de planifier la séquence d’actions pour réaliser sa mission (le déplacement). Cela se fait à partir de l’analyse de la perception.

L’action

consiste à mettre en œuvre des commandes ou des consignes aux actionneurs. Cette étape se fait généralement par des asservissements de bas niveau.

1.8.2 La navigation globale et la perception locale

1.8.2.1 La navigation globale

La navigation globale (l’approche délibérative) (figure 1.11) consiste à relier d’une manière séquentielle les trois actions : percevoir, planifier et agir. Après avoir perçus l’environnement, les systèmes de navigation peuvent être grossièrement classés en fonction des types de cartes utilisées : les cartes métriques, les cartes topologiques, les cartes hybrides et puis passer à la planification.

Dans cette étape, on parle de la planification globale qui cherche un chemin optimal et génère une trajectoire pour atteindre une cible [8].

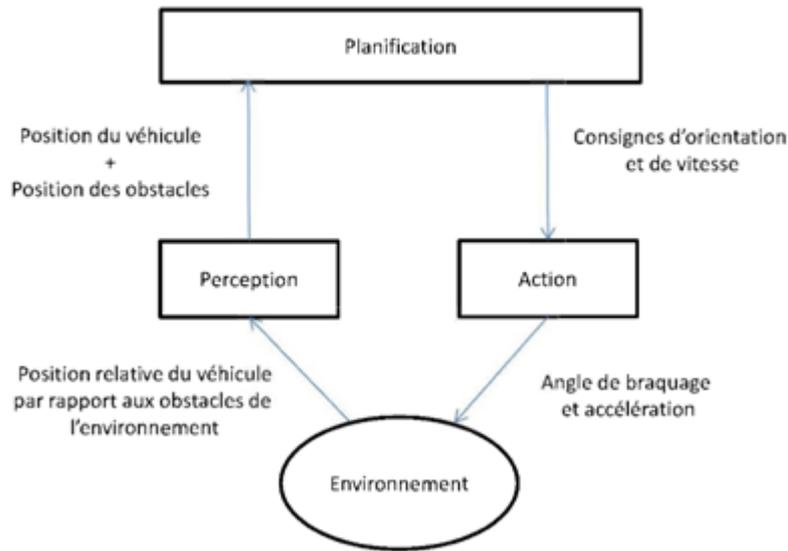


FIGURE 1.11 – navigation globale (délibérative)[8]

L'étape de planification nécessite une modélisation de l'environnement. Différentes représentations existent pour modéliser un environnement et seront présentés ci-dessus :

Les représentations de l'environnement

Pour permettre à un véhicule autonome de naviguer dans un environnement inconnu a priori, il faut construire une représentation de l'environnement à partir de la perception. Il est aussi très important dans une navigation autonome que le véhicule se localise par rapport à son environnement. La modélisation de ce dernier est donc indispensable [8]. Pour cela, on quatre types de cartes comme mentionné dans la figure 1.12 et expliqué ci-dessous :

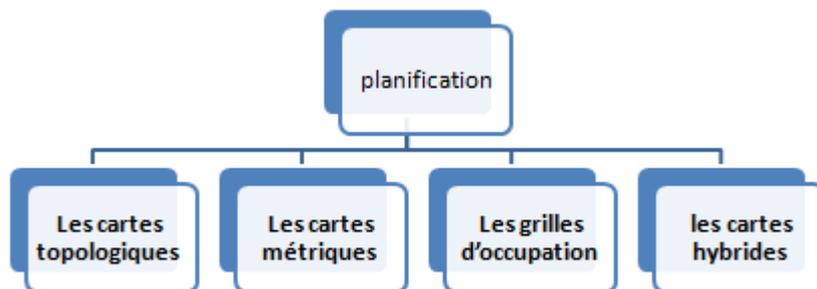


FIGURE 1.12 – modélisation de l'environnement de la navigation de véhicule autonome

- Les cartes topologiques :

Les cartes topologiques enregistrent un ensemble de lieux accessibles au véhicule ainsi que le moyen de passer d'un lieu à ses voisins sous forme d'un graphe. Une carte topologique mémorise un ensemble de régions (bureaux, couloirs, intersections, rues, champs...) ainsi que les liens qui existent entre ces régions (figure 1.12). Cette carte est représentée comme un graphe. Ce dernier est constitué de noeuds qui sont les régions (lieux) que le véhicule peut atteindre et d'arêtes qui permettent de lier les noeuds (chemins).

L'avantage de cette représentation est qu'elle ne requière pas un modèle métrique issu de la fusion de données de capteurs proprioceptifs. Les cartes topologiques ont l'avantage de fournir directement une segmentation de l'espace en différents lieux qui peut être intéressante pour la navigation ou l'interaction avec l'homme.

- Les cartes métriques :

Les cartes métriques permettent d'enregistrer dans un repère commun les obstacles ou les points de repère utilisés pour la navigation. La carte métrique représente les caractéristiques d'objets per (position, longueur, distance à un autre objet) dans un espace métrique (figure 1.12). Ce dernier est généralement exprimé en trois dimensions et définit un référentiel unique pour toutes les informations per. Les objets détectés dans cette carte correspondent aux obstacles que le véhicule pourrait rencontrer dans son environnement et représentent l'espace occupé. Le complémentaire correspond alors à l'espace navigable, c'est-à-dire, à l'espace dans lequel le véhicule peut se déplacer. Cette vision objective de l'environnement, plutôt indépendante de tout véhicule donné, facilite également la réutilisation de ces cartes par différents véhicules. Pour l'être humain, ces cartes sont plus faciles à comprendre car elles correspondent à sa propre perception. Les cartes métriques sont plus faciles à construire que les cartes topologiques en raison de la définition non ambiguë des lieux fournie par leurs coordonnées. Les cartes métriques ont l'avantage de fournir une estimation plus précise de la position.

- Les grilles d'occupation :

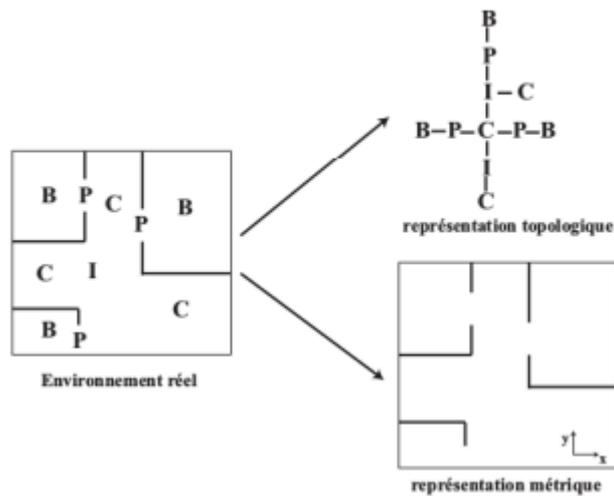


FIGURE 1.13 – les représentations de l’environnement : la représentation topologique mémorise des lieux et des liens de déplacement, dans cet exemple B=Bureau, P=Porte, C=Couloir, I=Intersection.les représentation métrique mémorise des objets (murs) avec une position dans un repère global

Les cartes basées sur les grilles d’occupation sont des représentations métriques discrétisées de l’environnement du véhicule. Elles représentent des environnements par des cellules métriques qui contiennent des informations relatives à l’occupation de l’environnement. Chaque cellule contient un indice qui indique si l’espace correspondant est plutôt libre ou occupé. Cette représentation requiert très peu de temps de calcul, ce qui revient à une mise à jour d’une carte rapide et facile. Le principe d’inférence bayésienne et les fonctions de croyance peuvent être utilisé pour estimer la probabilité d’occupation de chaque cellule.

- Les cartes hybrides :

La représentation basée sur les grilles produit des cartes métriques précises et peut être utilisée pour une localisation plus précise et un évitement d’obstacles. Cependant, si une représentation métrique est utilisée dans des environnements de grandes tailles, la complexité de l’espace et du calcul deviendra non négligeable. La représentation topologique peut être utilisée beaucoup plus efficacement pour la planification de chemins. La précision et la cohérence des données sont difficiles pour la carte topologique dans un environnement à grande échelle. Ainsi, la carte hybride qui combine les deux approches devient plus appro-

priée pour représenter l'environnement. Cela va permettre de surpasser les limitations de chaque représentation et d'exploiter les avantages de chacune. Les cartes hybrides sont des cartes topologiques qui contiennent des informations métriques sur les arêtes et les nœuds du graphe. De nombreuses approches hybrides ont également été développées pour tirer partie des deux approches et plusieurs systèmes récents utilisant des ensembles de poses relatives rendent leurs frontières relativement floues. Enfin, les méthodes de cartographie sémantiques intègrent hiérarchiquement ces types de cartes avec des informations supplémentaires telles que la nature des pièces ou les objets présents dans l'environnement.

1.8.2.2 Perception locale

La perception représente l'aptitude du système à extraire, étudier et mettre en œuvre des données acquises par des capteurs pour permettre au véhicule de prendre les décisions et agir en fonction de l'environnement qui l'entoure. Il existe deux types de perception : une globale et une locale. La perception locale désigne l'utilisation d'une information centrée sur le véhicule. Par exemple pour une perception visuelle, une seule image peut être prise. La perception globale désigne l'utilisation d'une séquence d'informations (une séquence d'images) et un repère de référence extérieur au robot.

La perception de l'environnement consiste à détecter les obstacles fixes et mobiles, etc. Le véhicule peut utiliser deux types de capteurs comme source d'informations : des capteurs proprioceptifs et des capteurs extéroceptifs. Le premier type (odomètres, gyroscopes, ...) fournit des données relatives à l'état interne du véhicule tandis que le deuxième type (Caméra, Laser, Lidar, Ultrasons, ...) fournit des informations sur l'environnement. Les données récupérées au cours de la navigation par les capteurs permettent de prendre des décisions, de planifier les trajectoires et de passer à l'action en fonction de l'environnement et les obstacles qui l'entourent [9].

1.8.3 La navigation des véhicules autonomes basée sur la communication V2X

La communication de véhicule avec tous (Vehicle-to-Everything V2X) est un système de communication véhiculaire qui permettra aux véhicules d'échanger des informations entre eux, avec les infrastructures et les piétons. Le V2X comprend d'autres types de communication plus spécifiques comme V2V (Vehicle-to-Vehicle), V2I (Vehicle-to-Infrastructure), V2R (Vehicle-to-Route), V2P (Vehicle-to-Pieton), V2S (Vehicle-to-Sensor) . Les buts principaux du V2X sont d'assurer la sécurité routière, de réduire l'embouteillage et de préserver l'environnement en économisant de l'énergie [10].

1.9 Conclusion

Dans ce chapitre, on a présenté et décrit les principaux concepts de l'IoV, ses caractéristiques, ses architectures et modèles de réseau et ses technologies. Ensuite, on a défini la véhicule autonome. Enfin, on a terminé par la navigation globale et perception locale des véhicules autonomes et la communication V2X.

Dans le prochain chapitre, on va présenter les entités de communication liés à la navigation autonome dans les environnements routière et de discuter quelques approches utilisés pour la communication V2X.

Chapitre 2

L'état de l'art

2.1 Introduction

Avec le nombre croissant de véhicules autonomes qui circulent sur la route, Les ITS les systèmes de transport intelligents ont vu le jour comme étant des systèmes composés de diverses entités, telles que les véhicules, les smartphones et PDA des piétons et les infrastructures. La nécessité d'une coopération entre ces entités est plus cruciale. Cette coopération passe par des mécanismes de communication appropriés entre tous ces entités sur la route pour communiquer et partager les informations.

Dans cette partie , on introduit les technologies de communication V2X qui sont utilisées dans les systèmes de véhicules autonomes, comme la communication V2V(Véhicule-à-Véhicule), V2P(Véhicule-à-Piéton), V2I(Véhicule-à-Infrastructure).

2.2 Les entités de communications

La communication entre les véhicules, entre les véhicules et les Piétons, ou entre les véhicules et les infrastructures est réalisée souvent à travers une technologie sans fil. Cette méthode de communication offre une vaste gamme d'informations aux conducteurs et aux voyageurs et permet aux applications de sécurité d'améliorer la sécurité routière et de fournir une conduite confortable.

Les principaux composants d'un système de communication véhiculaire sont une unité d'application AU (Application Unit), une unité à bord OBU (On Board Unit) et une unité coté route RSU (Side Road Unit). D'une manière typique, le RSU héberge une application qui fournit des services et l'OBU est un dispositif de pairs qui utilise les services fournis par la RSU.

L'application peut résider dans la RSU ou dans l'OBU, le dispositif qui héberge l'application est appelé le fournisseur, et le dispositif qui utilise l'application est décrit comme étant l'utilisateur. Chaque véhicule est équipée d'une OBU et d'un ensemble de capteurs pour collecter et traiter les informations, puis de l'envoyer en tant que messages à d'autres véhicules ou RSU à travers le support sans fil, elle comporte également une AU unique ou multiple qui utilise les applications fournies par le fournisseur en utilisant les capacités de connexion de l'OBU. La RSU peut également se connecter à Internet ou à un autre serveur qui permet de l-AU à partir de plusieurs véhicules de se connecter à Internet. La figure 2.1 illustre les différents types d'échanges de données utilisés dans la communication V2X.

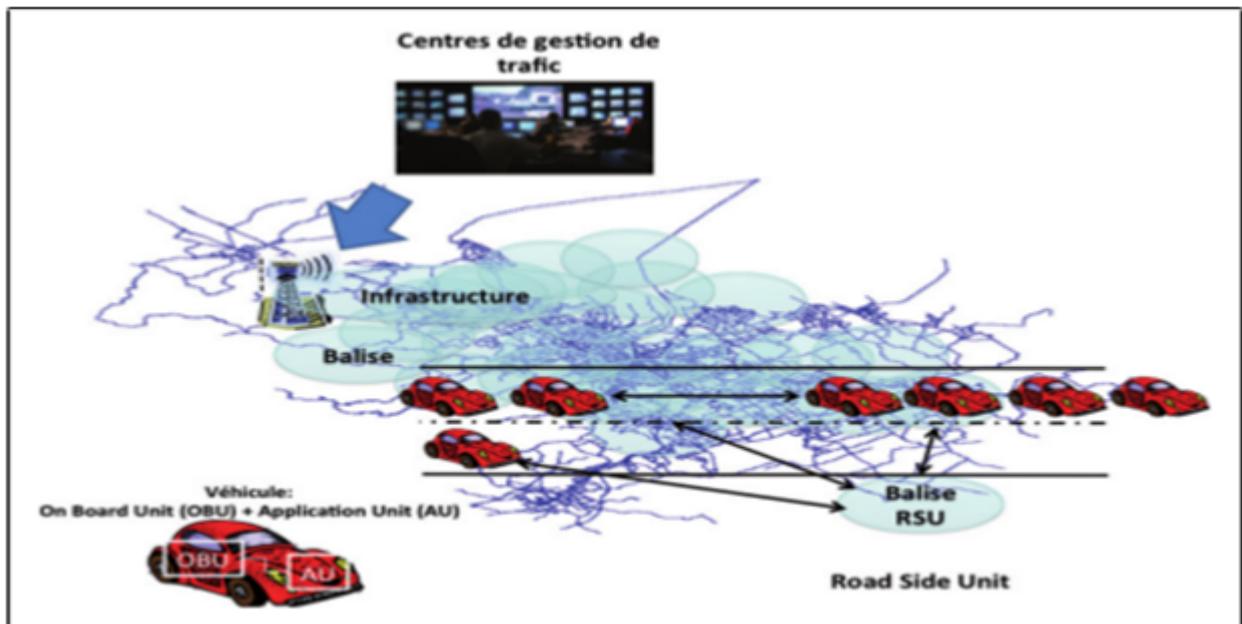


FIGURE 2.1 – Les entités de communications

2.2.1 unité à bord OBU(On Board Unit)

Une OBU (figure 2.2) est un dispositif à ondes généralement monté à bord d'un véhicule utilisé pour échanger des informations avec les unités d'action restreintes ou avec d'autres OBUs. Une OBU se compose d'un processeur de commande de ressource(RCP), les ressources comprennent une mémoire de lecture et d'écriture utilisée pour stocker et récupérer des informations, d'une interface utilisateur, une interface spécialisée pour se connecte aux d'autres OBU qui est un dispositif de réseau de communication sans fil de courte portée basée sur la norme IEEE 802.11p, elle peut comprendre une autre périphérique réseau pour les applications non sécurisées basées sur d'autres technologies radio telles que IEEE 802.11a/ b / g / n.

L'OBU se connecte à RSU ou à une autre OBU par une liaison sans fil sur la base du canal de fréquence radio IEEE 802.11p, elle est responsable de la communication avec les différentes RSUs et fournit également un services de communication aux données de l'AU et transmet les données d'autres OBU sur le réseau .

Les principales fonctions de l'OBU sont l'accès radio sans fil ad hoc, le routage géographique, le contrôle de la congestion du réseau, le transfert des messages fiable et la sécurité des données, la mobilité IP, d'assurer la localisation, la réception, le calcul, le stockage et l'envoi des données sur le réseau [11].



FIGURE 2.2 – On Board Unit(OBU)

2.2.2 unité d'application AU (Application Unit)

L'AU est un appareil embarqué dans le véhicule qui utilise les applications fournies par le fournisseur en utilisant des capacités de communication de l'OBU, l'AU peut être un dispositif dédié pour les applications de sécurité ou un dispositif normal comme un assistant numérique personnel (PDA) pour se connecter à Internet [11].

L'AU peut être connectée à l'OBU via une connexion filaire ou sans fil et peut résider avec l'OBU dans une seule unité physique, la distinction entre l'AU et l'OBU est logique. L'AU communique avec le réseau uniquement par l'OBU qui assume la responsabilité de toutes les fonctions de mobilité et de routage.

2.2.3 une unité côté route RSU(Road Side Unit)

La RSU (figure 2.3)est un dispositif d'onde fixe, située habituellement le long du bord de la route ou dans les endroits dédiés tels qu'aux intersections ou à proximité de places de stationnement. La RSU est équipée d'un dispositif de réseau pour une communication à courte portée dédiée basée sur la technologie de radio IEEE 802.11p, et peut également être équipée d'autres dispositifs de réseau de manière à être utilisés au fins de la communication au sein d'autres réseaux comme les réseaux d'infrastructures(figure 2.4 à 2.6). Chaque RSU à pour objectif de transmettre des messages aux véhicules qui se trouvent dans sa zone radio, ces messages contiennent des informations sur les conditions météorologiques, ainsi que sur l'état de la route(vitesse maximale autorisée, autorisation de dépassement, etc). Elle joue aussi le rôle d'un point d'accès afin d'assurer les communication V2I.



FIGURE 2.3 – Road Side Unit (RSU)

Selon le consortium de communication (C.C)(le consortium de communication c'est un groupe de réflexion), les principales fonctions et les procédures associées à une RSU sont [11] :

1. Extension de la portée de communication du réseau ad-hoc par redistribution les informations à d'autres OBU.

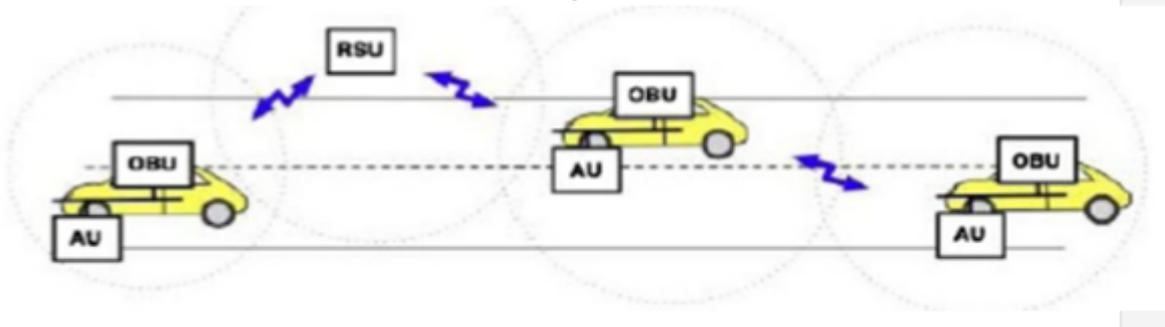


FIGURE 2.4 – RSU étend la portée du réseau ad-hoc en transmettant les données entre les OBUs

2. Applications de sécurité, comme un avertissement de pont bas, l'alerte d'un accident ou d'une zone de travail en utilisant l'infrastructure de communication de véhicule et agissant comme source d'information.

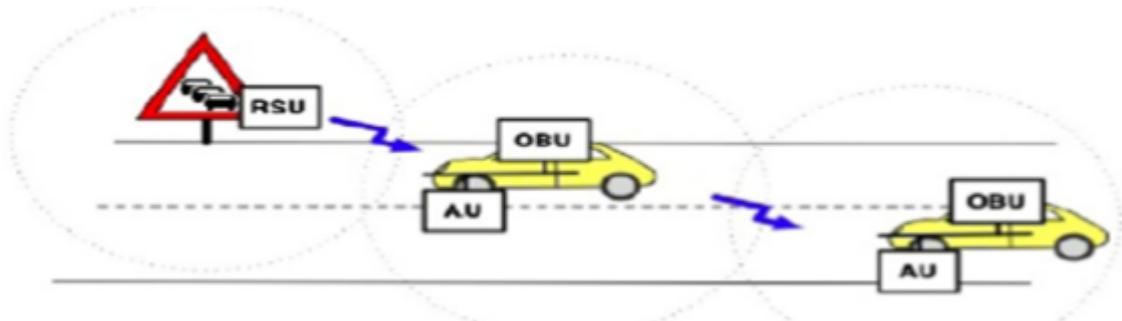


FIGURE 2.5 – fonctionne comme source d'information(exécution d'application de sécurité)

3. Fourniture d'une connectivité Internet aux OBU.

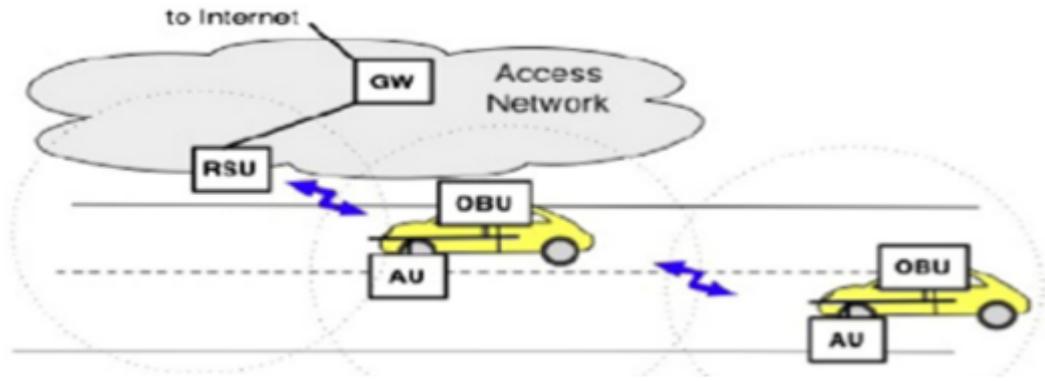


FIGURE 2.6 – RSU fournit une connectivité Internet aux OBUs

2.3 Les approches utilisées pour la communication V2X

La communication véhiculaire, appelée véhicule à tout (Vehicle to Everything V2X), contient généralement les cinq types suivants : communication véhicule à véhicule (V2V), communication véhicule à piéton (V2P), communication véhicule à infrastructure (V2I), communication véhicule à capteur (V2S) et communication véhicule à route (V2R). Cependant, ces cinq types de V2X peuvent utiliser la sensibilisation coopérative pour fournir des services plus intelligents aux utilisateurs finaux, pour la sécurité, l'amélioration du contrôle du véhicule autonome en utilisant les données de capteur à proximité et la prévention des accidents.

Cela signifie que les entités de transport, telles que les véhicules, les piétons et les infrastructures routières, peuvent collecter les connaissances sur leur environnement local (par exemple, les informations reçues depuis d'autres véhicules ou équipements de détection à proximité) pour traiter et partager ces connaissances afin de fournir des services plus intelligents. La conduite autonome assistée par la communication V2X génère un trafic de données sensible au retard avec les exigences de haute fiabilité. Ainsi, ces exigences peuvent être décrites comme une faible latence, une grande fiabilité, un trafic élevé et une mobilité élevée -au-delà de ce que les technologies sans fil actuelles peuvent fournir.



FIGURE 2.7 – trois modes de la communication V2X[12]

2.3.1 La navigation des véhicules autonomes basée sur la communication V2V

principe : La communication V2V(figure 2.8) est une communication entre les véhicules, ce mode de communication fonctionne à l'aide des OBU's. En effet, la communication entre deux véhicules se fait directement sans utiliser une infrastructure pour pouvoir communiquer entre eux. Dans l'interaction V2V, une véhicule communique directement avec une autre véhicule pour échanger les informations concernant le passager dans les véhicules, l'état des freins, la vitesse, la position du véhicule et les alertes de conduite dans la zone de couverture par un saut ou bien par le biais d'un protocole multi-sauts qui se charge de transmettre les messages de bout en bout en utilisant les nuds voisins qui les séparent comme des relais.

On mentionne comme un exemple de V2V, un véhicule qui marche en avance, il traverse une piste glacée et son système de freinage antiblocage (ABS) et / ou le système de contrôle électronique de la stabilité (ESC) s'enclenche. Les communications sans fil immédiates envoient des signaux d'avertissement aux autres véhicules pour ralentir ou faire un détour pour éviter cette route glacée. On mentionne aussi le cas d'un véhicule qui est impliqué dans un accident et que son système d'airbag est activé. Des signaux sans fil sont immédiatement envoyés aux véhicules dans la région pour réduire leurs vitesses ou se préparer à s'arrêter pour éviter les collisions de chaînes (les carambolages). Pour effectuer correctement cette mission V2V vitale, les communication sans

fil doivent présenter une latence très faible[13].

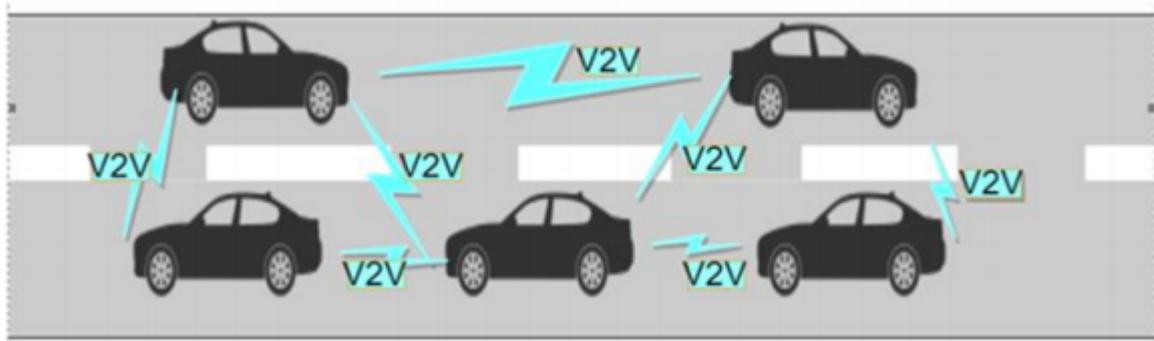


FIGURE 2.8 – mode de communication V2V

Avantages :

- Les communications V2V sont très efficaces pour le transfert des informations concernant les services liés à la sécurité routière
- Dans ce mode, les supports de communication utilisées sont caractérisés par un grand débit de transmission et une petite latence
- La connexion entre les véhicules est simple
- La communication V2V est utilisée pour partager les avertissements avant et après les collision, les conditions routières en temps réel, etc
- Dans ce mode, chaque véhicule est équipé pour communiquer directement avec un autre véhicule et cette communication est moins couteuse et plus flexible
- Elle permet d’optimiser l’ensemble du trafic routière et de réduire les embouteillages

Inconvénients :

- Les communications V2V ne garantissent pas la connectivité permanente entre les véhicules
- Celle-ci ne fonctionne qu’à courte distance

2.3.2 La navigation des véhicules autonomes basée sur la communication V2P

Principe : La communication V2P est une communication entre le véhicule et le piéton, ce mode de communication peut être averti d’un passage de piétons a fin de les protéger même

dans les conditions de faible visibilité comme dans une nuit sombre, dans un brouillard ou lors d'une forte pluie. Les appareils mobiles ou portables des piétons peuvent être utilisés pour la communication V2P. La communication V2P est une tâche essentielle pour la sécurité globale de toutes les personnes qui se présentent à la route. Dans le domaine des STI, les approches utilisées pour la sécurité des piétons sont divisées en deux catégories : celles basées sur les capteurs installés dans la route et celles basées sur la communication. Par exemple, on trouve des applications qui revent des informations des véhicules pour récupérer la localisation provenant des smartphones des piétons et identifier les situations de danger. De plus, les piétons peuvent utiliser des informations capturées par des capteurs de véhicules[13].

Avantages :

- L'utilisation des communications V2P améliore la perception de l'environnement pour la détection des piétons
- L'amélioration de la sécurité routière
- Les véhicules peuvent communiquer avec les piétons même lorsqu'ils ne sont pas en visibilité directe et sous des cas de faible visibilité tels que l'obscurité, fortes pluies, temps brumeux, etc

Inconvénients :

- La communication V2P dépend de la présence des piétons

2.3.3 La navigation des véhicules autonomes basée sur la communication V2I

Principe : La communication V2I (figure 2.9) est une communication entre un véhicule et une infrastructure réseau. Ce mode de communication est assuré grâce aux différentes entités du réseau VANETs. En effet, les OBUs des véhicules, les RSUs placés aux bords des routes et même les AUs contribuent tous entre eux pour assurer les communications V2I et pour fournir des services comme l'accès à internet[13].

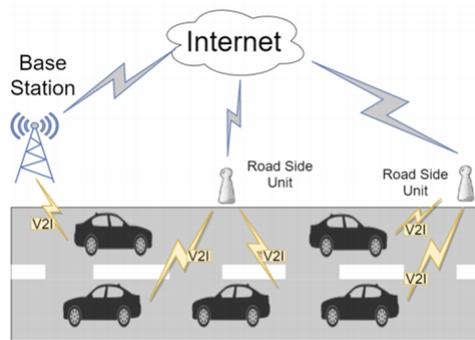


FIGURE 2.9 – le mode de communication V2I

Avantages :

- Ce mode de communication assure une connectivité relativement forte par rapport à la communication V2V
- Il assure une meilleure utilisation des ressources du réseau
- Il permet aux véhicules de bénéficier de plus de fonctionnalités et services comme l'accès à l'internet et les informations météorologiques
- La communication V2I améliore la perception de l'environnement et partage les informations utilisées qui améliorent la navigation du véhicule et la sécurité routière
- La communication V2I offre les détails du cycle des feux de circulation, les alertes potentielles de danger routière et autres informations contextuelles
- V2I fournit des informations comme l'espace de stationnement disponible, la congestion du trafic, l'état de la route, etc

Inconvénients :

- Ce mode est coûteux
- Dans la communication V2I, un nombre considérable de points d'accès doit être déployés dans l'environnement routier qui a un coût élevé

2.4 Conclusion

Dans ce chapitre, nous avons présenté les entités de communications. Ensuite un état de l'art sur quelques approches utilisées pour la navigation des véhicules autonome basée sur la communication V2X.

Dans le chapitre suivant, nous présentons la conception de notre système de navigation d'un véhicule autonome basée sur la communication V2I. Ceci peut contribuer à améliorer la perception de l'environnement et partager les informations utiles afin de renforcer la navigation et la sécurité routière.

Chapitre 3

Conception et développement

3.1 Introduction

Dans les chapitres précédents, nous nous sommes concentrés sur la partie théorique de notre projet qui nous a permis d'identifier la problématique et les concepts à prendre en considération dans la conception et la réalisation de notre système.

Ce chapitre sera consacré à la conception de ce système dédié à la navigation d'un véhicule autonome basé sur la communication V2X. Nous allons décrire l'architecture globale et détaillée ainsi que la méthode que nous avons suggérée pour planifier une trajectoire basée sur l'algorithme de Dijkstra et une méthode locale basée sur l'algorithme génétique afin de trouver un itinéraire alternatif dans un temps réduit. Enfin, nous expliquons la communication V2x du chemin découvert

3.2 La conception générale du système

Afin d'éclaircir notre travail, le schéma ci-dessous présente la conception générale qui montre que notre système a besoin de l'information suivante :

1. La carte d'environnement où va se déplacer le véhicule qui doit être préalablement définie.

Pour pouvoir à la fin :

1. Prendre la décision pour déterminer une série de mouvements appropriés à réaliser.
2. Tracer la trajectoire optimale entre le point de départ et la destination.

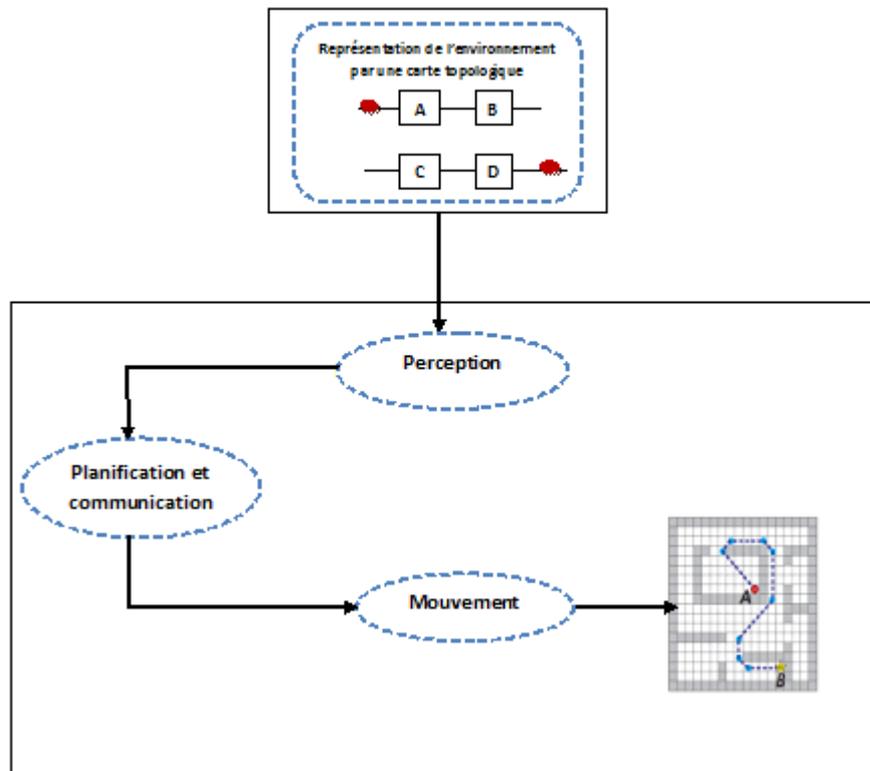


FIGURE 3.1 – Conception générale du système

3.3 La conception détaillée du système

Notre système passe par les étapes essentielles comme le montre en détail la figure 3.2, c'est pour générer un système de navigation d'un véhicule autonome basée sur la communication V2X dans un environnement de conduite.

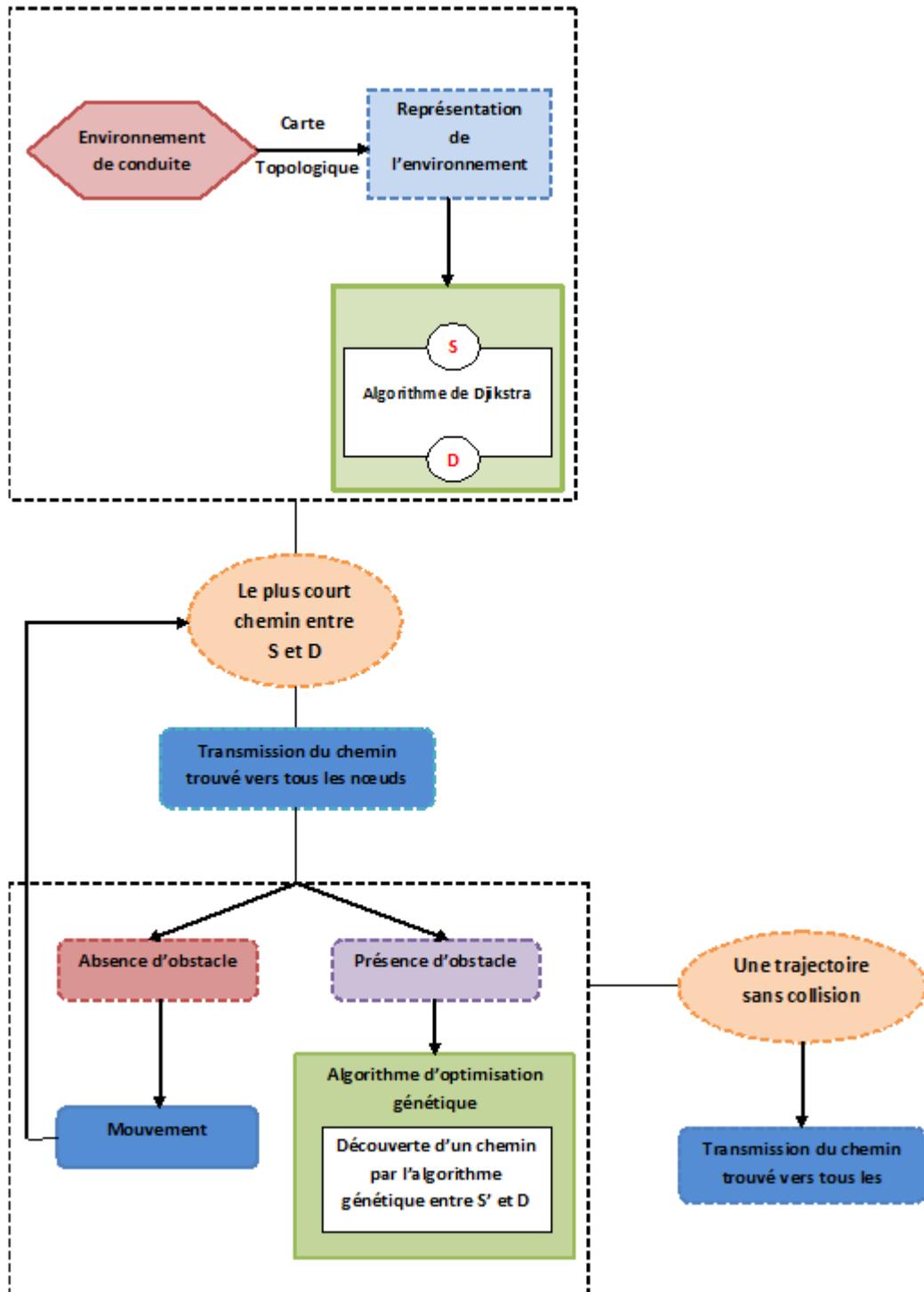


FIGURE 3.2 – Conception détaillée du système

3.3.1 Modélisation de l'environnement

La modélisation de l'environnement par une carte topologique va nous permettre d'avoir une représentation et une mémorisation de l'environnement sous forme d'un graphe, qui va nous

faciliter la recherche d'une trajectoire par l'utilisation d'un des algorithmes puissants de la théorie des graphes tel que l'algorithme de Dijkstra.

3.3.2 Algorithme de djikstra pour la navigation

1. **Principe :** L'algorithme de Dijkstra est un algorithme permet de trouver le plus court chemin entre le point de départ et le point de destination dans un graphe [14].

Voici le fonctionnement de l'algorithme de Dijkstra :

- (a) On construit un graphe G uniquement composé du point de départ
- (b) On fait la liste des arcs tels que :
 - i. Le point de départ appartienne à G
 - ii. Le point d'arrivée appartienne à G'
- (c) Si la liste est vide, on s'arrête(il n'existe pas de chemin entre les points considérés)
- (d) A partir de cette liste, on sélectionne l'arc qui minimise la distance entre le point de départ et le nouveau point
- (e) On ajoute cet arc à G' , ce qui ajoute également un point au graphe. On garde une trace du chemin ayant permis d'arriver au nouveau point
- (f) Si le point fraîchement ajouté correspond au point d'arrivée recherché, on s'arrête et on retourne le chemin complet(qui est le chemin le plus court)
- (g) Sinon, on retourne à (b)

2. Pseudo code de l'algorithme de Dijkstra :

Une fois que l'exécution de l'algorithme de Dijkstra est terminée, la planification du chemin le plus court entre la source et la destination est effectuée.

Algorithm 1 Pseudo code de l'algorithme Dijkstra

```
function Dijkstra(Graphe,source) :  
  dist[source] ← 0  
  for chaque sommet v dans Graphe do:  
    dist[v] ← infini  
    précédent[v] ← indefini  
  end for  
  Q ← l'ensemble de tous les noeuds du graphe  
  While Q n'est pas vide do  
    u ← sommet dans Q avec la plus petite dist[u]  
    if dist[u] = infini then  
      break  
    end if  
    supprimer u de Q  
    for chaque voisin v de u do  
      Alt ← dist[u] + dist_entre(u,v)  
      if alt < dist[v] then  
        dist[v] ← alt  
        précédent[v] ← u  
      end if  
    end for  
  end while  
  return dist[], précédent[]  
end function
```

3.3.3 Algorithme génétique pour contourner un obstacle

1. **Principe** : Les algorithmes génétiques (AG) sont des algorithmes de recherche et d'optimisation stochastique basés sur des mécanismes de la sélection naturelle et la génétique. Ils fonctionnent avec une population initiale regroupant un ensemble d'individus appelés chromosomes distribués sur l'entièreté de l'espace de recherche (toutes les solutions possibles)[14]. Chaque chromosome est constitué d'un ensemble de gènes. La performance relative de chaque individu est évaluée par une fonction objectif (appelée aussi de fitness) qui permet de créer une nouvelle population de solutions potentielles. Par la suite, les opérateurs évolutionnaires qui sont la sélection, le croisement et la mutation sont exécutés. On répète ce cycle jusqu'à ce que l'on trouve une solution satisfaisante[15].

2. **Pseudo code de l'algorithme génétique** :

Algorithm 2 Pseudo code de l'algorithme génétique

$P_0 \leftarrow \text{Population_initiale} ()$

$E_0 \leftarrow \text{Evaluer}(P_0)$

While laconditiond'arrêtn'estpasvérifiée **do**

$P \leftarrow \text{Sélection}(P_0)$

$P \leftarrow \text{Croisement}(P)$

$P \leftarrow \text{Mutation}(P)$

$P \leftarrow \text{Remplacer} (P_0, P)$

End while

return cheminOptimal

3. Schéma représentatif du fonctionnement de l'algorithme génétique :

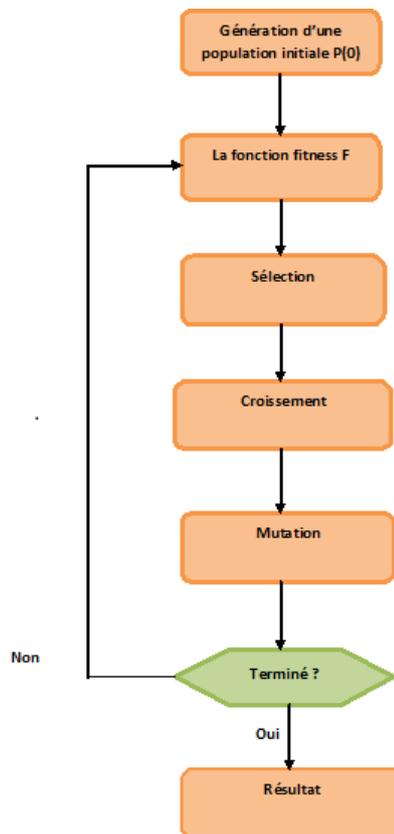


FIGURE 3.3 – Fonctionnement de l'algorithme génétique

4. Processus de l'algorithme génétique pour l'évitement de l'obstacle :

- Génération de la population initiale : Ce mécanisme doit être capable de produire une population d'individus non homogène qui servira de base pour les générations futures. Le choix de la population initiale est important car il peut rendre plus ou moins rapide la convergence vers l'optimum global. Une solution est un chemin de la source à la destination en passant par des points de relais.
- La fonction de fitness : Cette fonction prend ses valeurs dans \mathbb{R} et est appelée fonction d'évaluation de l'individu. Celle-ci est utilisée pour sélectionner et reproduire les meilleurs individus de la population. Dans conception, la fonction objectif est la distance euclidienne parcouru par le véhicule de la source à la destination.
- L'opérateur de sélection : La sélection permet d'identifier statistiquement les meilleurs solutions et d'éliminer les mauvais. les meilleurs solutions lié directement à la fonction de fitness.
- L'opérateur de croisement : Suite à la génération de la population initiale, nous procédons la phase de croisement qui assure la recombinaison de deux chemins parents pour former de nouveaux chemins enfants. Pour ce faire, nous considérons p la position de croisement, et nous traitons cette phase comme le montre la figure 3.5. Dans le but d'obtenir les meilleurs chromosomes.

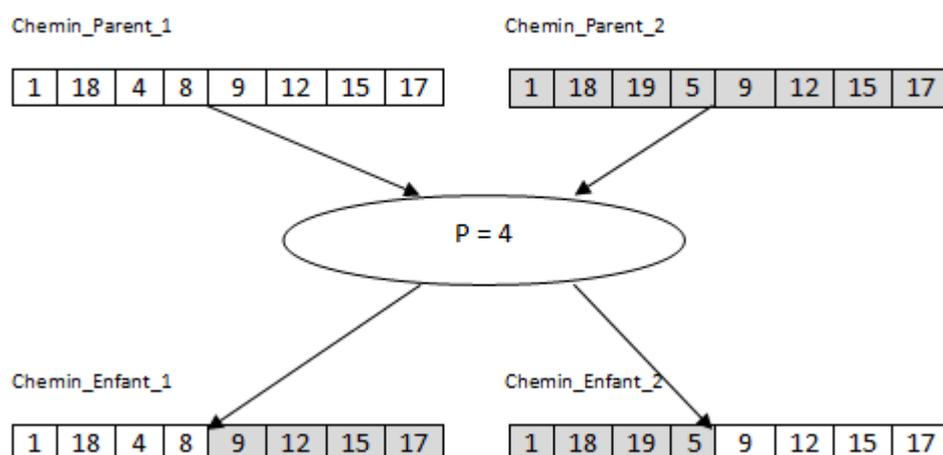


FIGURE 3.4 – Représentation de la Croisement

- L'opérateur de mutation consiste à choisir deux noeuds aléatoire dans une population et à échanger leurs valeurs respectives. La mutation ne permet pas l'obtention de meilleures solutions, mais elle permet de garder une diversité dans l'évolution des populations et d'éviter les optimums locaux, et se protège contre une perte irrécouvrable dans les caractéristiques des populations.

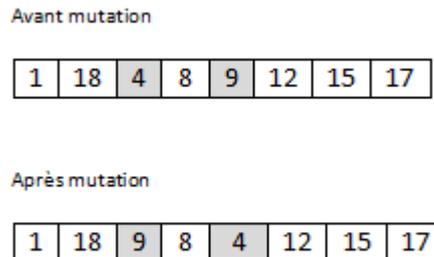


FIGURE 3.5 – Représentation de la mutation

3.3.4 Communication V2X du chemin découvert

Le véhicule peut échanger les données du chemin trouvé et transmettre ses informations (débit, délai, etc.) vers tous les noeuds de l'environnement routière grâce aux différentes interfaces réseaux telles que le WAVE/DSRC(Wireless Access in Vehicular Environments / Dedicated Short Range Communication), WIFI, 4/5G au via les différents points d'accès des RSU déployées aux bords des routes.

Plus précisément, la communication V2X est basée sur le DSRC (IEEE 802.11p) : Elle prend en charge la communication directe entre vcules (V2V) et entre véhicules et infrastructure (V2I). Les autres nuds du réseau peuvent par la suite profiter de cette information (nouvel itinéraire) pour une utilisation ultérieure.

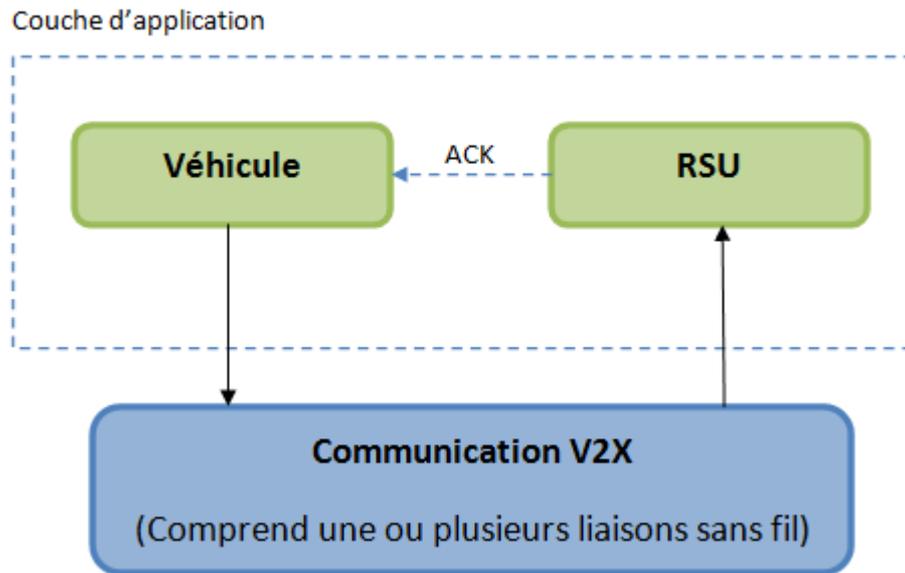


FIGURE 3.6 – modèle de la communication V2X

3.4 Conclusion

Dans ce chapitre, nous avons décrit notre système avec une conception générale et détaillée. Nous avons détaillé la méthode proposée (l'algorithme génétique) pour résoudre le problème de la planification du chemin qui permet au véhicule d'atteindre sa destination en évitant les obstacles qui pourraient se trouver sur ce chemin.

Dans le chapitre suivant, nous allons présenter l'étude expérimentale afin de valider cette étude, on illustre l'implémentation du modèle on va aussi décrire les différents composants nécessaires à son fonctionnement, l'environnement de développement. A la fin, nous présenterons et interpréterons les différents résultats obtenus par l'algorithme génétique en les comparant avec l'algorithme de Dijkstra considéré comme une méthode de planification traditionnelle.

Chapitre 4

Implémentation et résultats

4.1 Introduction

Afin de réaliser notre projet, nous avons eu recours à plusieurs outils de développement dans le côté logiciel. Ce chapitre est consacré à la présentation de l'environnement de travail, le langage de programmation et les outils utilisés pour démontrer le fonctionnement de notre système.

Ce chapitre présente aussi les codes de programmation des algorithmes proposés qu'on a développé tel que l'algorithme de Dijkstra, l'algorithme génétique, les résultats obtenus qui sont comparés à l'algorithme de navigation traditionnel Dijkstra. Les résultats et les comparaisons sont exposés afin de montrer l'efficacité de notre proposition pour planifier une trajectoire optimale dans un temps très réduit et pour transmettre efficacement cette découverte à tous les nœuds du réseau.

4.2 Environnement d'expérimentation

Pour l'implémentation de notre système nous avons utilisé :

4.2.1 Environnement matériel

Durant le développement de notre approche, on a utilisé un ordinateur "lenovo" possédant les caractéristiques suivante :

- Processeur : intel (R) core(TM) i5-4202Y CPU @ 1.60GHz * 4
- Mémoire : 4 GO
- Système d'exploitation : ubuntu 16.04 LTS, 64-bit

4.2.2 Environnement logiciel

Langage de programmation Python 2.7 :

La programmation de notre projet a été faite en langage Python développé et créé au début des années quatre-vingt-dix par Guido Van Rossum au Centrum voor Wiskunde à Informatica, à Amsterdam au Pays-Bas.



Python est un langage de programmation objet interprété, ce qui veut dire que les instructions qu'on lui envoie sont transcrites en langage machine au fur et à mesure de leur lecture, c'est un langage puissant, à la fois facile à apprendre et riche en possibilités.

Le simulateur NS-2 :

NS-2 (Network Simulator 2) est un simulateur de réseau à événements discrets et open source. Il est utilisé pour la simulation des protocoles de réseau avec différentes topologies. Il est capable de simuler des réseaux câblés et sans fil.

NS-2 a été construit en C++ et fournit une interface de simulation via OTcl. Dans ns-2, l'animateur de réseau (NAM) est utilisé pour la vue graphique du réseau. NS-2 est le simulateur de réseau le plus répandu et le plus utilisé pour les travaux de recherche. NS-2 nécessite deux langages : OTcl (Object Tools Command Language) et C++. A travers le langage OTcl, l'utilisateur d'écrit les conditions de la simulation : topologie du réseau, caractéristiques des liens physiques, protocoles utilisés etc. Ainsi, Le langage C++ sert à décrire le fonctionnement interne des composants de la simulation[17].

- Flot de simulation : NS-2 interprète le script OTCL et exécute la simulation. Les résultats obtenues (Fichier trace et fichier NAM) peuvent être visualisés avec l'outil NAM (Network Animator) et analysés à partir des courbes tracées par l'outil Xgraph (Un traceur de graphes).

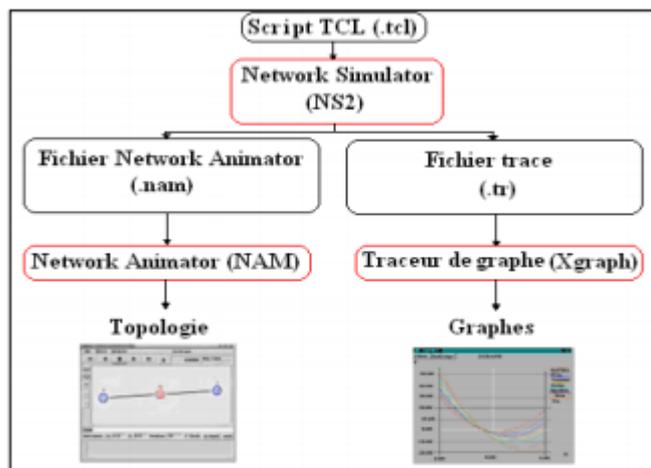


FIGURE 4.1 – Flot de simulation avec NS-2

1. NAM : Le NAM est un outil d'animation utilisé pour interpréter et visualiser le trace de simulation des réseaux et le trace de données.
2. Fichier trace : Le fichier trace est un fichier qui retrace tous les évènements qui se sont produits pendant la simulation à savoir émission, réception, perte de paquets.
3. AWK : AWK est un langage pour le traitement des fichiers de texte, inspirée du langage

C. Chaque fichier est traité comme une séquence d'enregistrements, et par défaut, chaque ligne est un enregistrement. Il décrit comment filtrer le fichier de sortie .tr de NS-2 pour extraire des données significatives de la simulation.

4.3 La navigation communicante de véhicule autonome par un algorithme génétique :

4.3.1 Modélisation de l'environnement de navigation :

pour conduire des tests sur la navigation de véhicules autonomes basée sur la communication V2X, un modèle de mobilité doit être défini qui présente un environnement routier correspondant à un quartier par exemple. Ce modèle doit définir les intersections de l'environnement, les routes, les RSU, le point de départ et la destination du véhicule. Pour notre étude, nous avons opté pour un environnement routier simulé avec le simulateur NS-2 (expliqué en détail dans la section 4.4).

4.3.2 Développement et implémentation de la navigation globale :

Nous avons développé et implémenté l'algorithme de Dijkstra qui retourne l'itinéraire le plus court qui relie le point de départ et la destination en passant par des noeuds intermédiaires.

Code de l'algorithme Dijkstra

```

def dijkstra(graph,start,goal):
    shortest_distance = {}
    predecessor = {}
    unseenNodes = graph
    infinity = 9999999
    path = []
    for node in unseenNodes:
        shortest_distance[node] = infinity
    shortest_distance[start] = 0
    while unseenNodes:
        minNode = None
        for node in unseenNodes:
            if minNode is None:
                minNode = node
            elif shortest_distance[node] < shortest_distance[minNode]:
                minNode = node
        for childNode, weight in graph[minNode].items():
            if (weight + shortest_distance[minNode]) < shortest_distance[childNode]:
                shortest_distance[childNode] = weight + shortest_distance[minNode]
                predecessor[childNode] = minNode
        unseenNodes.pop(minNode)

    currentNode = goal
    while currentNode != start:
        try:
            path.insert(0,currentNode)
            currentNode = predecessor[currentNode]
        except KeyError:
            print('Path not reachable')
            break
    path.insert(0,start)
    if shortest_distance[goal] != infinity:
        print('Shortest distance is ' + str(shortest_distance[goal]))
        print('And the path is ' + str(path))

```

Fonctionnement

L'algorithme prend en entrée un graphe pondéré par des réels positifs et un noeud source. Il s'agit de construire progressivement un sous-graphe dans lequel sont classés les différents noeuds par ordre croissant de leur distance minimale au sommet de départ. La distance correspond à la

somme des poids des arcs empruntés.

Au départ, on considère que les distances de chaque noeud au noeud de départ sont infinies, sauf pour le noeud de départ pour lequel la distance est nulle. Le sous-graphe de départ est l'ensemble vide.

Voici un exemple de recherche du chemin le plus court entre A et J en appliquant Dijkstra :

```
graphe = [  
  ( 'A' , 'B' , 8 ),  
  ( 'A' , 'C' , 14 ),  
  ( 'A' , 'E' , 11 ),  
  ( 'B' , 'F' , 7 ),  
  ( 'C' , 'G' , 13 ),  
  ( 'C' , 'H' , 9 ),  
  ( 'H' , 'D' , 12 ),  
  ( 'H' , 'J' , 10 ),  
  ( 'F' , 'I' , 15 ),  
  ( 'I' , 'J' , 6 ),  
  ( 'E' , 'J' , 30 )  
]
```

- Étape 01 : on choisit le noeud A. on met à jour les noeuds voisines de A qui sont B, C et E. leurs distances deviennent respectivement 8, 14, 11
- Étape 02 : on choisit le nud B qui est à distance minimale(8). on met à jour le seul voisin de B et le noeud F .alors, sa distance devient $8+7 =15$
- Étape 03 : on choisit F. on met à jour le voisin I avec la distance $15+15=30$
- Étape 04 : on choisit E. on met à jour le voisin J avec la distance $30+6=36$.
- Étape 05 : la distance la plus courte en dehors du sous-graphe est maintenant celle de le nud C. on choisit donc C, on met à jour le nud G($14+13=27$) et le noeud H($14+9=23$).
- Étape 06 : la distance la plus courte en dehors du sous-graphe est maintenant celle de le nud H. on choisit donc H(23), on met à jour le nud D($23+12=35$) et le noeud J($23+10=33$) .alors, $33<35$.
- étape 07 : la distance la plus courte suivante est celle le noeud G. on choisit donc G, la mise à jour ne change aucun autre distance.

- Étape 08 : la distance la plus courte suivante est celle de le nud I. la distance de le noeud voisin J n'est pas modifiée car la distance existante est inférieure à celle que l'on obtiendrait en passant par I($30+6 > 33$)
- Étape 09 : le nud dont la distance est la plus courte est J (33) . on choisit J et on l'ajoute au sous-graphe. On s'arrête puisque la ville d'arrivée est maintenant dans le sous-graphe.

En neuf étapes, on peut déterminer le chemin le plus court entre le noeud A et J Chemin = [A , C , H , J] , avec la distance est 33.

4.3.3 Développement et implémentation de la navigation locale :

Le véhicule exécutera l'algorithme génétique pour régénérer un autre chemin alternatif.

Code de l'Algorithme génétique

```
def genetic_algorithm(self, gen_max, pop_size):
    gen = 1
    self.population_size = pop_size
    self.generate_population(pop_size)
    if not quite:
        pretty_print('Initital:')
        self.print_chromosomes(self.population)

    while gen <= gen_max:
        gen += 1
        p = 1
        new_population = list()
        while p <= self.population_size:
            p += 1
            parents = random.sample(range(self.population_size), 2)
            newbie = self.crossover(self.population[parents[0]], self.population[parents[1]])
            newbie.mutate()
            fit = self.fitness(newbie)
            self.results.append((newbie, fit))
            new_population.append(newbie)
            if self.best is None or self.best[1] > fit:
                self.best = (newbie, fit)

        if not quite:
            pretty_print('%dth generation (after crossover, mutations): ' % gen)
            self.print_chromosomes(new_population)

        self.selection(self.population, new_population)
        if not quite:
            pretty_print('After selection: ')
            self.print_chromosomes(new_population)

    return self.best
```

Fonctionnement

Nous fixons la taille de la population à $N = 4$. Le Nombre d'itération (critère d'arrêt) = 16

— Nous appliquons la méthode de la génération de la population

```
def generate_population(self, pop_size):  
    path=[]  
    all_path.g.printAllPathsUtil(self.source, self.destination, [False]*(19), path)  
    all=list()  
    chromosomes = list()  
    all=all_path.equalize(19)  
    for e in all:  
        print(e ,self.fitness(Chromosome(e)))  
  
    for i in range(pop_size):  
        chromosomes.append(Chromosome(all_[random.randint(0, len(all_)-1)]))  
  
    print(chromosomes)  
  
    self.population = chromosomes
```

Cette méthode permet de générer un population de 4 chemins possible non homogène pour les générations futures.

— Nous appliquons la méthode de La sélection

```
def selection(self, prev, now):  
  
    prev.extend(now)  
    prev.sort(key =lambda x: self.fitness(x))  
    self.population = prev[:self.population_size]
```

Cette méthode permet d'identifier statistiquement les meilleurs solutions et d'éliminer les mauvais pour créer une nouvelle population à partir de l'ancienne. les meilleurs solutions lié directement à la fonction de fitness.

— Ensuite, nous appliquons la méthode de La croisement

```

def crossover(self, mother, father):
    mother_list = mother.get()
    father_list = father.get()
    cut = random.randint(0, self.chromosome_length - 1)
    child = mother_list[0:cut] + father_list[cut:]
    return Chromosome(child)

```

Cette méthode fait la recombinaison de deux chemins parents pour former de nouveaux chemins enfants, dans le but d'obtenir les meilleurs chemins.

— nous appliquons la méthode de la mutation

```

def mutate(self):
    pos = random.randint(1, self.size() - 2)
    self.vector[pos] = random.randint(0, self.size() - 1)

```

Cette méthode permet de choisir deux nœuds aléatoires dans une population et faire échanger de leurs valeurs respectives.

4.3.4 la communication V2X du chemin découvert

la communication est responsable de la transmission des données entre le nœud de départ et la destination afin de trouver la meilleure solution.

4.4 Expérimentation et résultats

4.4.1 Scénarios d'une expérimentation

Pour valider notre proposition, nous avons proposé un scénario de test, et nous avons choisi comme intersection de départ le nœud '1' et comme destination l'intersection représenté par le nœud '17'. On utilise 19 nœuds et 22 chemins, comme le montre dans la figure 4.2 suivantes :

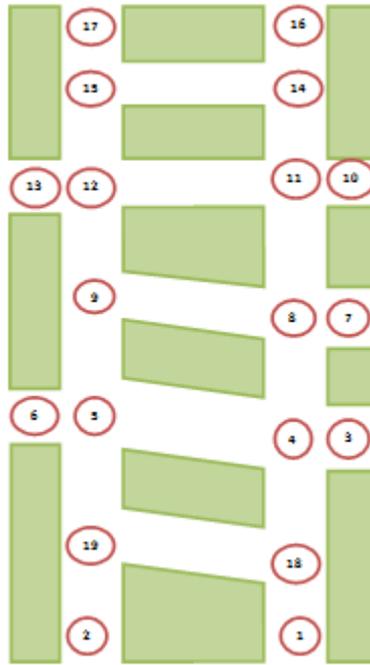


FIGURE 4.2 – Modèle de mobilité du véhicule

Les paramètres détaillés de ce scénario sont illustrés dans le tableau suivant :

Identifiant du nœud	Nœud suivant (Voisin)	Distance entre les deux nœuds
1	18	120
4	3	150
2	19	150
5	6	150
4	8	170
5	9	170
4	5	302
9	12	150
8	9	302
8	11	200
11	14	170
14	16	120
11	12	300
12	15	170
14	15	300
15	17	120
8	7	150
11	10	150
12	13	150
19	5	170
18	19	302

4.4.2 Résultats obtenus

Après une exécution de notre système sur le scénario proposé et pour donner suite à la navigation globale assurée par l'algorithme de Dijkstra, nous avons obtenu l'itinéraire suivant (en orange) avec une distance totale de 1222 cm.

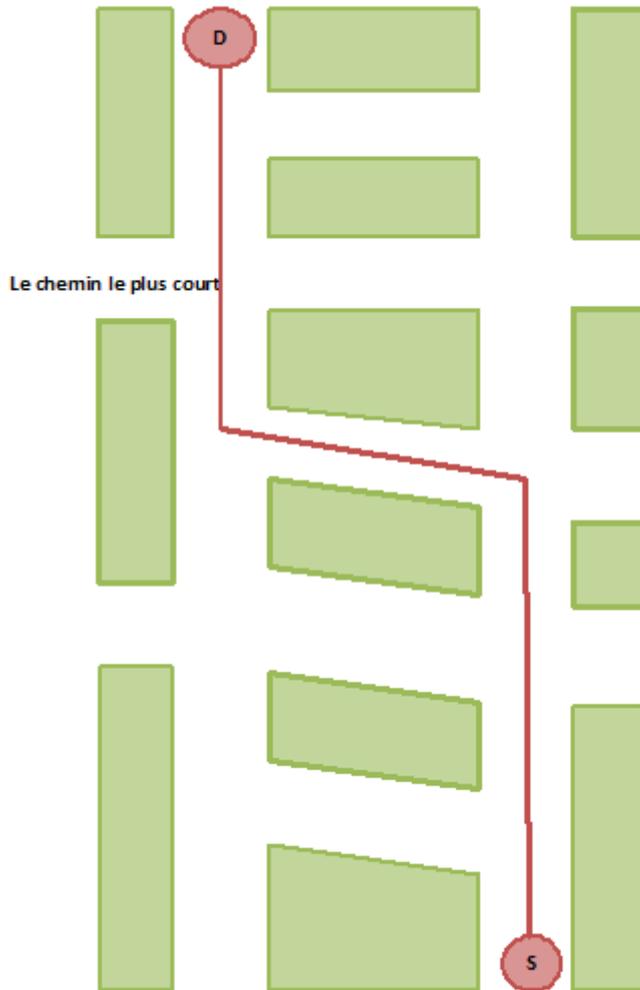


FIGURE 4.3 – L'itinéraire le plus court entre le point de départ et la destination trouvé par l'algorithme de Dijkstra

Après une exécution de notre système sur le scénario proposé et pour donner suite à la navigation locale assurée par l'algorithme génétique, nous avons obtenu l'itinéraire suivant (en orange) avec une distance totale de 1222 cm.

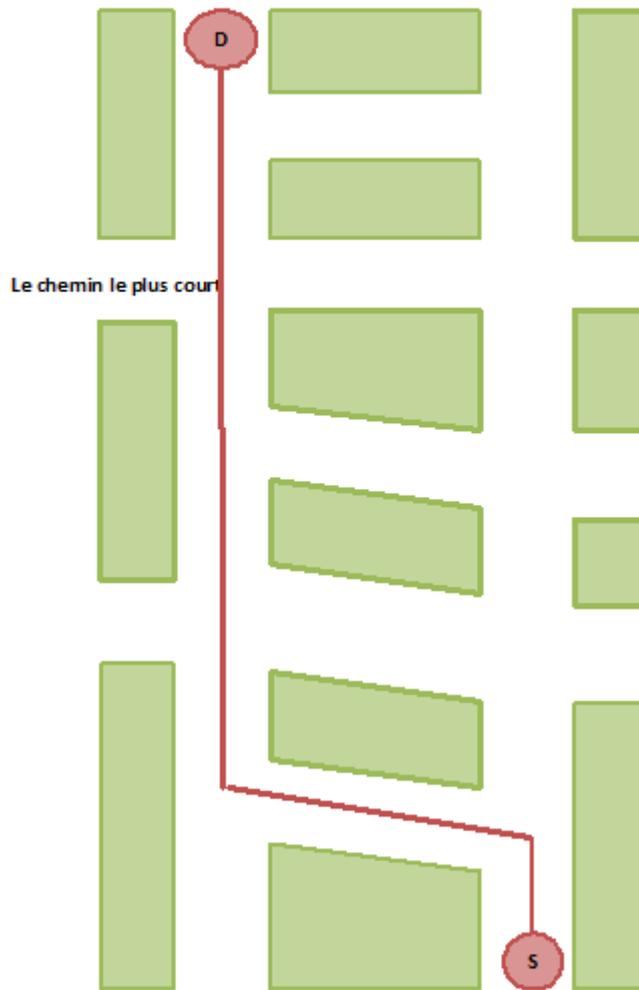


FIGURE 4.4 – L’itinéraire le plus court entre le point de départ et la destination trouvé par l’algorithme génétique

Pour montrer la performance de notre système, nous avons comparé les résultats obtenus avec l’algorithme de Djisktra et les résultats obtenus avec l’algorithme génétique.

Table 4.1 Tableau comparatif entre l’algorithme de Djisktra et l’algorithme génétique

Mesure \ Algorithme	Algorithme de Dijkstra	Algorithme génétique
Itinéraire	['1','18', '4', '8', '9', '12', '15', '17']	['1','18', '19', '5', '9', '12', '15', '17']
Distance	1222	1222
Complexité	$O(N^2)$	$O(N)$

4.5 Conclusion

Dans ce chapitre, nous avons présenté l'étude expérimentale afin de valider notre étude, nous avons décrit les différents composants matériels ou outils logiciels nécessaires au fonctionnement de notre projet, ensuite nous avons développé et implémenté la méthode que nous avons proposé dans le troisième chapitre.

A la fin, nous avons interprété les différents résultats obtenus par l'algorithme génétique, en les comparants avec l'algorithme de Dijkstra considéré comme une méthode de planification traditionnelle exacte.

Conclusion générale

Dans ce projet, nous avons développé un système de navigation du véhicule autonome par la planification d'une trajectoire qui permet au véhicule d'atteindre sa destination et transmise les informations avec les informations avec la RSU qui pourraient se trouver sur son chemin.

Notre système est basé sur une méthode que nous avons proposée, qui combine entre les deux types de navigation globale et locale. Dans la première étape, nous avons utilisé une méthode de planification globale basée sur l'algorithme de Dijkstra et dans la deuxième, une méthode locale basée sur l'algorithme génétique afin de trouver un itinéraire alternatif dans un temps très réduit.

Le véhicule peut échanger les données du chemin trouvé et transmettre ses informations (débit, délai, etc.) vers tous les noeuds de l'environnement routière grâce aux différentes interfaces réseaux telles que le WAVE/DSRC(Wireless Access in Vehicular Environments / Dedicated Short Range Communication), WIFI, 4/5G au via les différents points d'accès des RSU déployées aux bords des routes.

Nous avons commencé par la présentation du concept de l'internet des véhicules, et nous avons détaillé les concepts de base de la navigation autonome basée sur la communication V2X. Ensuite, nous avons présente quelques approches utilisées pour la communication V2X. La conception de notre système a été établie en illustrant la méthode proposée pour résoudre le problème de la planification d'une trajectoire entre le point de départ et la destination dans un temps réduit.

A la fin, nous avons exposé la réalisation des objectifs suivants notre système, en présentant l'environnement routier et quelque méthode pour le développement et l'implntation de notre système et les différents résultats obtenus.

Bibliographie

- [1] Indu, Sibaram Khara, "Internet of Vehicles (IOV) : Evolution, Architectures, Security Issues and Trust Aspects", International Journal of Recent Technology and Engineering (IJRTE), March 2019, p.268-269
- [2] Asiya Khan, Sanjay Sharma, Cindy Goh, Cheng Leong Lim, "A Review of Network Models for Internet of Vehicles", VEHICULAR 2017 : The Sixth International Conference on Advances in Vehicular Systems, Technologies and Applications, p.7
- [3] Abdus Samad, Shadab Alam, Mohammed Shuaib, Mohammad Ubaidullah Bokhari, "Internet of Vehicles (IoV) Requirements, Attacks and Countermeasures", conference paper in IEEE Conference, march 2018, p.4037-4038
- [4] OMPRAKASH KAIWARTYA, ABDUL HANAN ABDULLAH, YUE CAO, AYMAN ALTA-MEEM, MUKESH PRASA, CHIN-TENG LIN, XIULEI LIU, "Internet of Vehicles : Motivation, Layered Architecture, Network Model, Challenges, and Future Aspects", IEEE Access, 28/10/2016, p. 5360-5362
- [5] Alexia Biglia, "Analyse prospective sur l'implantation de la voiture autonome : impact sur l'industrie automobile et le citoyen", Th de doct. UNIVERSITE CATHOLIQUE DE LOUVAIN, 2014-2015
- [6] Society of Automotive Engineers. <https://www.sae.org/>. Accessed :10/11/2019
- [7] David ALBERT, Maha FASSI, Jean-Frans ROBIN, Sinwen STAMBOULI ,Auren TOUTAIN ,Louis ULMER, "Vehicule Autonome : Etat de l'art, capture et traitement de l'information embarquées", institut national des sciences applique de rouen, 19/06/2017

- [8] Hela Ben Said, "Navigation autonome et commande rrencapteurs de robots d'assistance et la personne", Th de doct. Universit Limoges, 23/03/2018
- [9] Adrien Durand-Petiteville."Navigation rrenculti-capteurs d'un robot mobile en environnement encombr. Th de doct. Universitul Sabatier-Toulouse III, 2012.
- [10] Jean Lecordier, Florent Marmin, Hugo Simoes Maricato, Clnt Lefebvre, Lisa Picard, Tairi Zheng, "Vcule Autonome : Sysys cooptifs/Communication", institut national des sciences appliqu de rouen, 17/06/2018
- [11] Saif Al-Sultan, Moath M.Al-Doori, Ali H.Al-Bayatti, Hussein Zedan, "A comprehensive survey on vehicular Ad Hoc network",in : ELSEVIER, 2013, p. 2-3
- [12] Hui Sheng Ma, Erqing Zhang, Shufang Li, Zhengnan Lv and Jing Hu, "A V2X Design for 5G Network Based on Requirements of Autonomous Driving",in : Society of Automotive Engineers (SAE), 09/14/2016,
- [13] Andras Kokuti, Ahmed Hussein, Pablo Marin-Plaza, Arturo de la Escalera, Fernando Garcia, "V2X Communications Architecture for Off-Road Autonomous Vehicles"
- [14] Wansu Lim, Lyle Parungao, Fabian Hein. "DIJKSTRA ALGORITHM BASED INTELLIGENT PATH PLANNING WITH TOPOLOGICAL MAP AND WIRELESS COMMUNICATION". In : ARPN Journal of Engineering and Applied Sciences 8.13 (2018), p. 27542763
- [15] Richa Mahajan, Gaganpreet Kaur , "Neural Networks using Genetic Algorithms", International Journal of Computer Applications, September 2013,p .6-7
- [16] LAETITIA JOURDAN, "mheuristiques pour l'extraction de connaissances :application gmi-que", Th de doct, universits sciences et technologies de lille, 26/11/2003]
- [17] Aina Randrianarisain,"Modsation de la consommation drgie En vue de la conception conjointe (matel/logiciel) des applications embarqu. Application aux raux de capteurs sans fil (wsn)", Th de Doct, 25/02/2015