

République Algérienne Démocratique et Populaire

Ministère de l'enseignement supérieur et de la recherche scientifique

Université de 8 Mai 1945 – Guelma -

Faculté des Mathématiques, d'Informatique et des Sciences de la matière

Département d'Informatique



Filière : Informatique

Option : STIC

Conception d'un modèle de contrôle adaptatif du trafic

Encadré Par :
Dr Benhamza Karima

Présenté par :
Bentadjine Amel

Juin 2022

Dédicace

J'ai le grand plaisir de dédier ce travail :

A mon très cher père « Lamine » :

Pour tes encouragements, ton soutien et surtout pour ton amour et ton sacrifice afin que rien m'entrave le déroulement de mes études.

A ma très chère mère «Nora » :

Quoi que je fasse ou que je dise, je ne saurais point te remercier comme il faut. Ton affection me couvre, ta bienveillance me guide et ta présence à mes côtés a toujours été ma source de force pour affronter les différents obstacles.

A mon cher frère «Rabah El Amine » :

Pour ton soutien moral et ton encouragement tout au long de mes études.

A ma très chère amie «Manar » :

Merci pour tout ma chérie, tu es vraiment une sœur.

A tous les membres de ma famille :

Et surtout ma tante «Nassima » et mes cousines « Ferial », « Ramla », « Choubaila » et « Meryem » qui m'ont supporté dans les moments difficiles.

Sans oublier mes chères amies

Nouha, Youssra, Khouloud et Hiba.

Remerciements

Je remercie en premier lieu ALLAH qui m'a éclairé mon chemin pour achever ce modeste travail.

Je tiens à remercier sincèrement mon encadreur Dr Benhamza Karima pour son encadrement, sa patience, sa gentillesse et ses orientations durant le semestre.

Je remercie aussi tous ceux qui ont, de près ou de loin, aidé à rendre ce travail possible, que ce soit par des informations, des idées ou par des encouragements.

Résumé

De nos jours, le transport routier est devenu un élément essentiel dans le fonctionnement des sociétés modernes. Donc la gestion d'un tel réseau est devenue importante. Parmi les outils les plus utilisés pour la gestion de ces réseaux, on trouve les feux de signalisation. Ces derniers ne s'adaptent pas avec la quantité du trafic (temps fixe pour chaque feu). L'apparition des nouvelles technologies a permis de résoudre ce problème et permet aussi aux feux d'être intelligents.

L'objectif principal de notre travail est de proposer une nouvelle solution de contrôle dynamique de feux de signalisation intelligente en utilisant l'apprentissage par renforcement combiné avec l'apprentissage profond. L'avantage de notre système est qu'il assure l'adaptation entre les feux de signalisation et la fluidification du trafic dans différentes conditions.

Mots clés : Signalisation intelligente, Contrôle de trafic, Apprentissage par renforcement, Apprentissage profond.

Abstract

Nowadays, transport has become an essential element for the modern societies. So the management of networks has become also important. Among the most used tools for the management of these networks, we find traffic lights. These lights do not adapt to the amount of traffic (fixed time for each traffic light). The evolution of new technologies has made it possible to solve this problem and to make traffic lights smart.

The objective of this work is to propose a new dynamic control solution for intelligent traffic lights using reinforcement learning combined with deep learning. The main advantage of our system is to provide adaptation between traffic lights and smooth traffic flow in different conditions.

Keywords: Smart traffic lights, Traffic control, Reinforcement learning, Deep learning.

ملخص

في الوقت الحاضر، أصبح النقل البري عنصرًا أساسيًا في عمل المجتمعات الحديثة. لذلك أصبحت إدارة مثل هذه الشبكة مهمة. من بين الأدوات الأكثر استخدامًا لإدارة هذه الشبكات نجد إشارات المرور. هذه الأخيرة، لا تتكيف مع مقدار حركة المرور (وقت محدد لكل إشارة مرور). أدى ظهور التقنيات الجديدة إلى حل هذه المشكلة وسمح للأضواء بأن تكون ذكية.

الهدف الرئيسي من عملنا هو اقتراح حل جديد للتحكم الديناميكي في إشارات المرور الذكية باستخدام التعلم المعزز والتعلم العميق أيضا. الميزة الرئيسية لنظامنا هي أنه يوفر التكيف بين إشارات المرور وتسهيل حركة المرور في ظروف مختلفة.

الكلمات المفتاحية: إشارات المرور الذكية، التحكم في حركة المرور، التعلم المعزز، التعلم العميق.

Table de matière

Dédicace	ii
Remerciements	iii
Résumé	iv
Abstract	v
ملخص	vi
TABLE DE MATIERE	VII
LISTE DES TABLEAUX	XII
INTRODUCTION GENERALE	1
Chapitre 1. Concepts de base	3
1. Introduction	4
2. Réseau routier	4
a. Route	4
b. Intersection	5
3. Congestion routière	6
3.1. Définition	6
3.2. Types de congestion	6
3.2.1. Congestion récurrente	6
3.2.2. Congestion non-récurrente	6
3.3. Conséquences de la congestion	6
4. Gestion du réseau routier	7
4.1. Système Trafic Intelligent (STI)	7
4.1.1. Historique	7
4.1.2. Définition	7
4.1.3. Services fournis par les STI	7

4.2.	Signalisation intelligente	9
4.2.1.	Feux tricolores	10
4.2.2.	Inconvénients des feux tricolores	11
4.2.3.	Définition de la signalisation intelligente	11
4.2.4.	Architecture générale de la signalisation intelligente	11
4.3.	Défis dans la conception de la signalisation intelligente	12
4.4.	Avantages de la signalisation intelligente	12
5.	Conclusion	13
Chapitre 2. Etat de l'art		14
1.	Introduction	15
2.	Travaux reliés	15
2.1.	Modèles mathématiques	15
2.2.	Modèles issus de l'intelligence artificielle	16
3.	Conclusion	20
Chapitre 3. Conception et implémentation		21
1.	Introduction	22
	I. Modélisation et conception	22
1.	Concepts clés de la modélisation	22
1.1.	Processus de Décision de Markov (PDM)	22
1.2.	Apprentissage par renforcement	22
	a. Q-learning	23
	b. Politique décisionnelle	24
1.3.	Apprentissage profond	24
2.	Modèle Proposé	25
2.1.	Modélisation	25
	a. L'agent	25
	b. Environnement	25
	c. L'état perçu par l'agent	25
	d. L'action	25
	e. La récompense	27
2.2.	Modèle du Q-learning amélioré par le Deep-learning	28
3.	Modèle de simulation proposé	29

II.	Implémentation et expérimentation	30
1.	Matériel utilisés pour le développement de l'application	30
2.	Outils logiciels utilisés pour le développement de l'application	31
3.	Exécution et résultats	31
3.1.	Paramètres du Q-Learning	31
3.2.	Paramètres du Deep-Learning	31
3.3.	Tableau d'initialisation des paramètres	32
4.	Déroulement de l'exécution	32
5.	Discussion des résultats	35
6.	Conclusion	38
	CONCLUSION GENERALE	39
	REFERENCES BIBLIOGRAPHIQUES	40

Liste des figures

Figure 1. Réseau routier à la ville de Yaoundi	4
Figure 2. Zones d'une intersection.....	5
Figure 3. Présentation de la structure du service télépéage.....	8
Figure 4. Avertissement pour la sécurité des véhicules	8
Figure 5. Présentation du service d'appel d'urgence	9
Figure 6. Signalisation intelligente en Algérie	9
Figure 7. Feu vert.....	10
Figure 8. Feu orange	10
Figure 9. Feu rouge.....	10
Figure 10. Structure de la signalisation intelligente.....	12
Figure 11. Structure d'apprentissage par renforcement	23
Figure 12. Environnement.	25
Figure 13. Vague Nord-Sud.....	26
Figure 14. Vague Nord-Sud gauche.	26
Figure 15. Vague Est-West.....	27
Figure 16. Vague Est-West gauche.	27
Figure 17. Modèle proposé.....	29
Figure 18. Intersection modélisée.....	30
Figure 19. Lancement de l'entraînement.....	32
Figure 20. Modèle en cours d'entraînement.....	33
Figure 21. Fin d'entraînement.	33
Figure 22. Présentation de l'intersection sur SUMO	34
Figure 23. Simulation en cours d'exécution.....	34

Figure 24. Fin de la simulation	35
Figure 25. Temps d'attente cumulé pour la phase d'entraînement. ...	35
Figure 26. Queue pour un faible débit (100 véhicules).....	36
Figure 27 Queue pour un trafic moyen (500 véhicules).....	37
Figure 28. Queue pour un haut débit (1000 véhicules).	37

Liste des tableaux

Tableau 1. Synthèse des travaux sur la signalisation intelligente	20
Tableau 2. Synthèse des travaux sur l'apprentissage par renforcement	21
Tableau 3. Matériel utilisé.....	30
Tableau 4. Paramètre du Q-Learning	31
Tableau 5. Paramètres du Deep-Learning	32
Tableau 6. Paramètres d'initialisation.....	32

Introduction générale

La croissance de la population et l'augmentation du nombre d'utilisateurs sur les routes forment une source considérable à plusieurs problèmes. Par exemple, un conducteur qui fréquente beaucoup les embouteillages, perd son temps et fait face toujours au stress qui peut impliquer directement aussi bien les accidents qu'indirectement, les problèmes de santé. Aussi la vie des utilisateurs est en risque à cause des rejets de gaz et des émissions polluantes causées par les embouteillages.

Donc le problème principal est la congestion qui est devenue une préoccupation majeure pour les spécialistes du transport.

Des feux de circulation classiques ont été mis en place pour gérer ce phénomène. En effet, la plupart des villes africaines en général et algériennes en particulier utilisent ce type de signalisation (temps fixe pour chaque feu) réglée une seule fois en tenant compte de prévisions du trafic. On peut observer par exemple que la durée des feux est la même à 8H00 du matin, où la circulation est complètement embouteillée par les camions de marchandises traversant la ville et les particuliers se rendant à leur travail, qu'à 22H00 le soir où la plupart des rues sont complètement dégagées [1]. Le minutage prédéfini ne tient pas compte des conditions du trafic.

On voit bien là, la nécessité de systèmes plus dynamiques connus sous le nom de systèmes de transport intelligent. Ces systèmes, permettent le contrôle adaptatif du trafic à l'aide des informations fournies par ses capteurs.

Dans ce travail, on s'intéresse à la proposition de système de contrôle des feux de circulation intelligent en utilisant une hybridation entre l'apprentissage par renforcement et l'apprentissage profond. L'objectif est de fluidifier le trafic et d'éliminer le problème de congestion.

Ce mémoire est structuré en trois chapitres :

Le premier chapitre présente les notions de base du trafic routier. Puis, il expose les systèmes de gestion de ce réseau en soulignant la signalisation intelligente.

Dans le deuxième chapitre, on expose les travaux existants sur la signalisation intelligente.

Dans le dernier chapitre, on présente la modélisation et l'implémentation de notre système, suivis des résultats obtenues.

On termine avec une conclusion générale, des perspectives futures de ce travail et des références bibliographiques.

Chapitre 1. Concepts de base

1. Introduction

La croissance rapide de l'urbanisation a entraîné une augmentation du nombre de véhicules dans les villes et les agglomérations. Par conséquent, de grandes préoccupations sont entraînées, telles que la congestion, les retards de transport et la pollution. C'est pourquoi, des nouvelles techniques de gestion de trafic sont apparues et qui s'insèrent sous le concept de "Systèmes du Trafic Intelligent".

Dans ce chapitre on va parler des notions de base du réseau routier et les systèmes de gestion de ce réseau en soulignant l'importance de la signalisation intelligente.

2. Réseau routier

On peut définir un réseau routier comme un ensemble d'intersections, liées entre eux avec des routes où chaque route comporte une ou plusieurs voies [2].

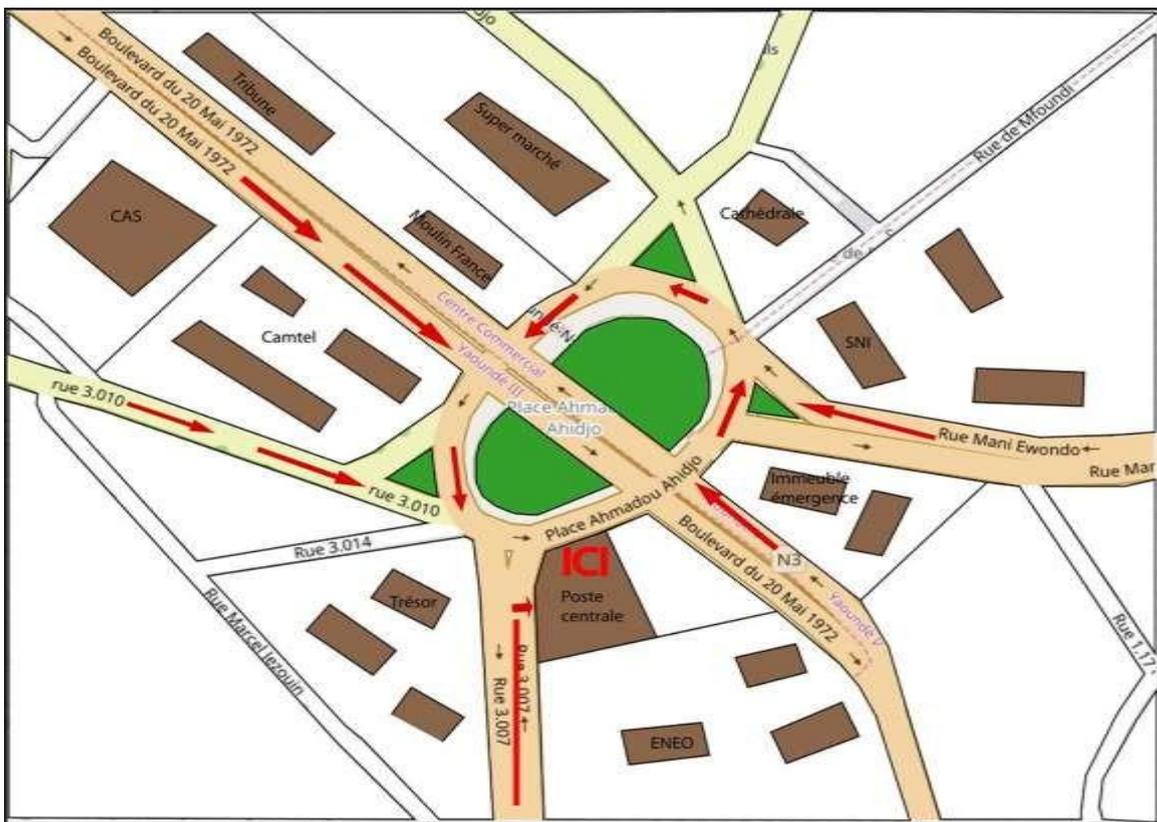


Figure 1. Réseau routier à la ville de Yaoundi [3].

a. Route [2]

C'est le trajet pour déplacer d'un endroit vers un autre. Elle se caractérise par sa longueur, nombre de voies et les directions de la circulation à l'intérieur.

b. Intersection

Le croisement entre deux routes ou plus forme une intersection [4]. Les mouvements au sein d'une intersection sont caractérisés par ses routes d'origine et ses routes de destination. Elle peut avoir plusieurs schémas à savoir la géométrie de l'infrastructure. Elle comporte trois zones [2] :

- Zone de conflit : elle présente la ressource critique que tous les véhicules partagent. C'est l'espace de croisement des routes.
- Zone de stockage : on peut la définir comme l'entrée de la zone de conflit.
- Zone de sortie : c'est la zone qui peut libérer l'intersection, elle forme la sortie de celle-ci.

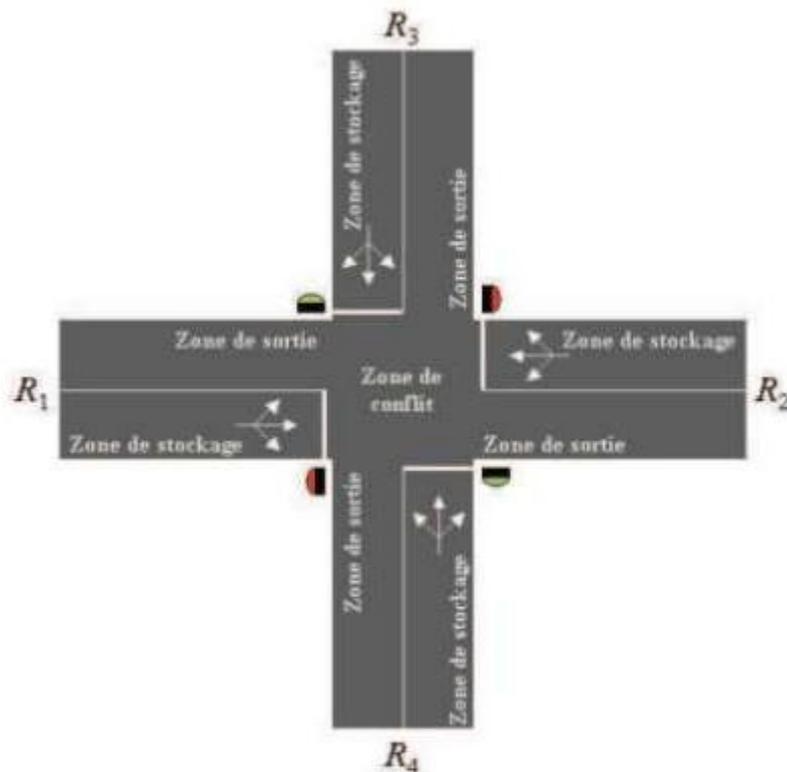


Figure 2. Zones d'une intersection [2].

3. Congestion routière

3.1. Définition

C'est un phénomène apparu lorsque la capacité du réseau routier s'approche de sa valeur maximale. On peut dire que la congestion est un cas où la demande du trafic dépasse l'offre [5]. Ce phénomène apparaît le plus souvent à des heures de pointe. Ces derniers représentent la période de grande fréquentation de la journée pendant laquelle un trafic est le plus important. Généralement, c'est le moment où la majorité des personnes actives se rendent à leur lieu de travail (entre 7h et 9h le matin) et puis lorsqu'ils rentrent à leur domicile (entre 17h et 20 h).

3.2. Types de congestion [6]

On a deux types de congestions :

3.2.1. Congestion récurrente

C'est la congestion liée au problème d'infrastructure, c'est à dire, la capacité d'une infrastructure conçue à un certain moment pourrait être inférieure à la demande réelle à un moment ultérieur. Même si cette capacité satisfait un certain niveau de demande, ce type de congestion va apparaître dans les heures de pointe.

3.2.2. Congestion non-récurrente

Ce type de congestion est lié aux perturbations sur le réseau routier. Ces perturbations peuvent être prévisibles comme les travaux sur les routes. Certaines d'autres sont moins prévisibles comme les accidents.

3.3. Conséquences de la congestion [7]

La congestion entraîne de nombreuses conséquences néfastes telles que :

- Le gaspillage des grandes quantités des produits pétroliers.
- La pollution.
- Empêchement des véhicules d'urgence.
- Perte du temps à cause des embouteillages.
- Augmentation du stress pour les automobilistes.

4. Gestion du réseau routier

La gestion du trafic routier s'inscrit dans le domaine des systèmes de transport intelligent [2].

4.1. Système Trafic Intelligent (STI)

4.1.1. Historique [8]

Dans les années 60 à 70, le secteur de l'informatique a connu des développements technologiques, à la suite desquelles les systèmes du trafic intelligent STI sont apparus. Parmi les premiers projets lancés on cite : le projet CACS (Comprehensive Automobile traffic Control System) au Japon, et aussi le système ALI (Autofahrer Leit und Information System) en Allemagne. Le but de ces projets était l'aide à la conduite en utilisant un ordinateur central qui calcule l'itinéraire effectué à chaque intersection.

Dans les années 80, et avec l'évolution de la technologie de nouveaux projets ont été proposés. Ils avaient les mêmes buts qui sont la réduction de la pollution, l'optimisation de la sécurité, etc. Les deux projets qui ont été développés sont (PROMETHEUS – PROgraM for European Traffic with Highest Efficiency en 1986 ; IVHS America – Intelligent Vehicle Highway Society of America en 1988). Le développement technique et technologique avec les objectifs attendus, forment ainsi une source pour l'innovation des nouveaux STI.

4.1.2. Définition

La notion d'intelligence des systèmes du trafic est liée à l'intégration des appareils et de la technologie de l'Internet des Objets (Internet Of Things - IOT) au sein de ces systèmes, tels que les capteurs, les actionnaires et les systèmes embarqués avec microprocesseur. En effet, les villes sont devenues plus intelligentes grâce au développement de l'IOT, qui a mené à la création des nouvelles applications visant à améliorer la vie des gens [9].

Les STI (ou bien en anglais ITS) permettent le traitement, l'analyse et de la communication des informations relatives au système de transport. Dans ces systèmes, les informations captées jouent un rôle important dans son fonctionnement de base principalement la communication et l'adaptation aux conditions [8].

4.1.3. Services fournis par les STI [10]

Les services des STI les plus fréquemment déployés sont :

- **Télépéage** : C'est un système de péage électronique conçu pour la facilitation des entries-sorties pour les conducteurs sur les routes exigent un paiement.

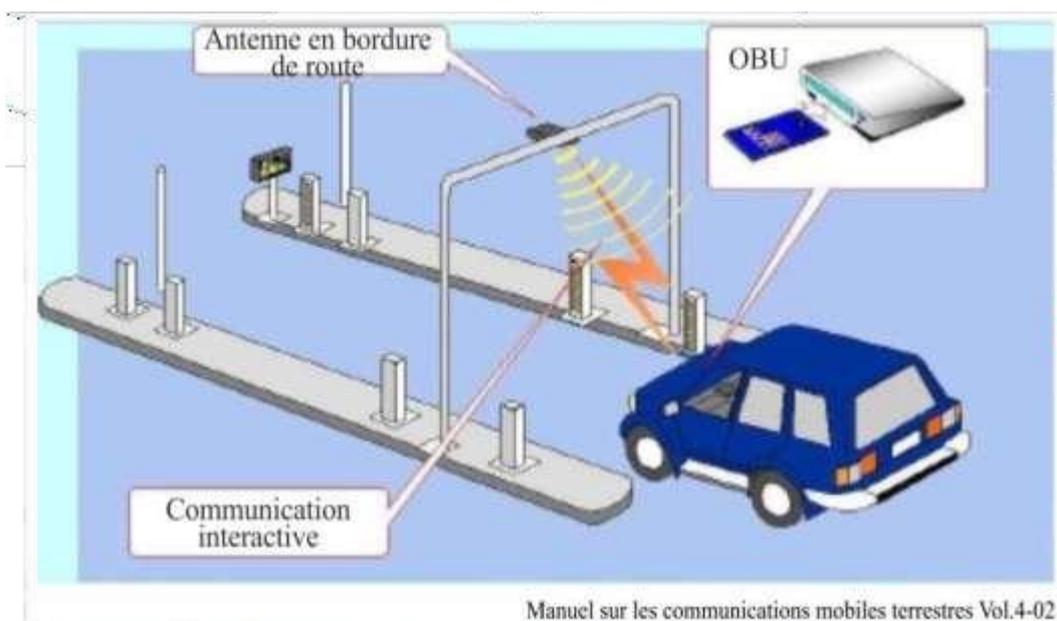


Figure 3. Présentation de la structure du service télépéage [10].

- Sécurité des véhicules et des routes : C'est le service qui aide à éviter les accidents de la route. Il permet aux véhicules de communiquer leurs positions afin d'éviter les collisions.

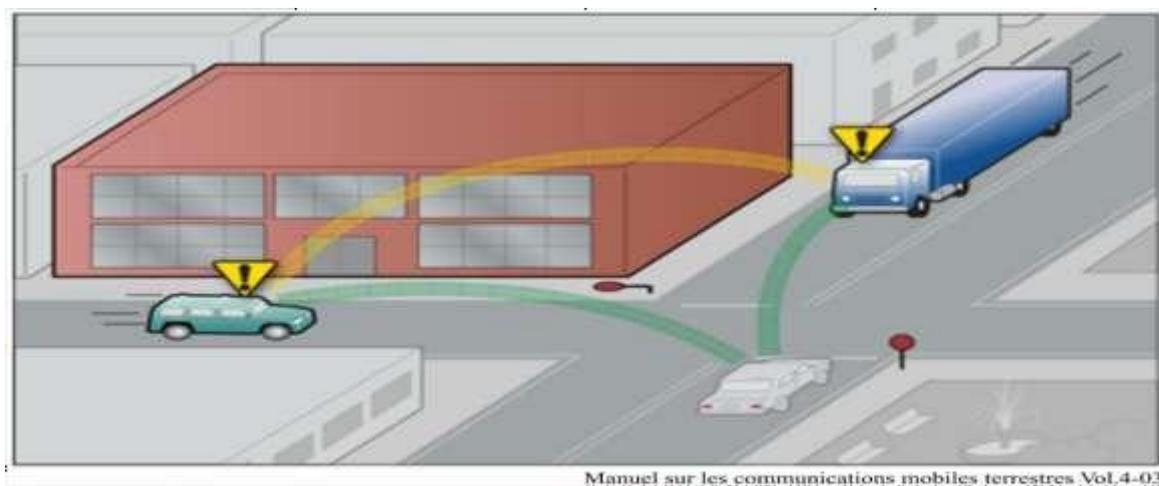


Figure 4. Avertissement pour la sécurité des véhicules [10].

- Appels d'urgence : C'est un service de signaux vocaux qui permet à un véhicule accidenté d'effectuer un appel d'urgence pour demander une intervention rapide.

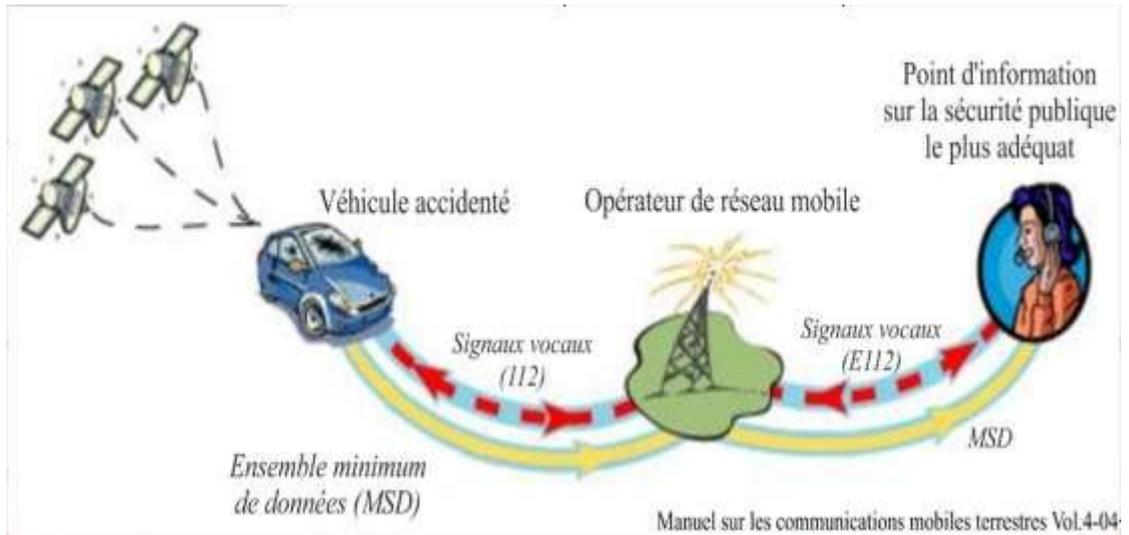


Figure 5. Présentation du service d'appel d'urgence [10].

- Signalisation intelligente : Adaptation du fonctionnement des feux de circulation tricolores avec le type de flux de voiture existant. C'est le système qui nous intéresse dans notre projet.



Figure 6. Signalisation intelligente en Algérie [11].

4.2. Signalisation intelligente

Avant d'exposer la signalisation intelligente, on doit d'abord définir les feux tricolores :

4.2.1. Feux tricolores

Aux niveaux des carrefours, les feux de circulation ou bien feux tricolore, sont des dispositifs aidant à la régulation du trafic routier entre les véhicules et les piétons [3]. Généralement, ils fonctionnent avec la séquence (vert, orange, rouge) avec un temps fixe pour chaque couleur [11].

- **Feu vert** : Celui qui indique que l'utilisateur a la propriété à procéder [3].



Figure 7. Feu vert [11].

- **Feu orange ou jaune** : Celui qui indique que le conducteur doit marquer l'arrêt, sauf s'il ne peut pas arrêter son véhicule dans les conditions de sécurité suffisantes [3].



Figure 8. Feu orange [11].

- **Feu rouge** : Celui qui indique au conducteur qu'il doit marquer absolument l'arrêt [3].



Figure 9. Feu rouge [11].

- **Un cycle de feu** :

C'est une séquence de phases [12].

- **Une phase de feu :**

C'est une durée pendant laquelle un ou bien plusieurs flux de véhicules cohérents (ensemble de véhicules qui peuvent franchir l'intersection au même temps) sont admis à l'intersection [12].

4.2.2. Inconvénients des feux tricolores [2]

- L'émission des gaz carboniques due aux files d'attente créées par les feux de signalisation non adaptatifs.
- L'augmentation des concentrations des particules du monoxyde du carbone et d'oxyde d'azote.
- La perte du temps.

4.2.3. Définition de la signalisation intelligente [11]

Les systèmes de signalisation intelligentes se caractérisent par l'ajout des capteurs intelligents, qui permettent de compter le nombre de véhicules au cours du temps et de fournir des informations sur le trafic routier (fluide, moyen, congestion). Avec ces informations, le système intelligent fait une régulation temporelle des feux afin d'assurer la fluidité du trafic.

4.2.4. Architecture générale de la signalisation intelligente

Afin de faciliter et fluidifier le trafic en diminuant le temps d'attente des véhicules, l'architecture générale de la signalisation intelligente du trafic aux intersections doit comporter deux modules importants :

- **Module de collecte des statistiques de trafic :** ce module décrit les technologies utilisées pour recueillir des informations sur l'activité courante du trafic sur les chemins [10]. Ces informations contiennent principalement le nombre de véhicules, la longueur de la file d'attente, la largeur de la route, etc... Ces technologies sont constituées par des capteurs de différents types. On cite ci-dessous les plus utilisés [9] :
- Boucles électromagnétiques.
- Capteurs vidéo (caméra intelligente).
- Réseaux de capteurs sans fil.
- Capteurs à hyperfréquence (radar).
- Fibres optiques.

- **Module de contrôle du trafic** : le contrôle intelligent des feux de signalisation (ou de circulation) a besoin d'une conception de contrôleur efficace, pour commuter, selon le trafic, les feux tricolores en utilisant les informations collectées par le module précédent [13].

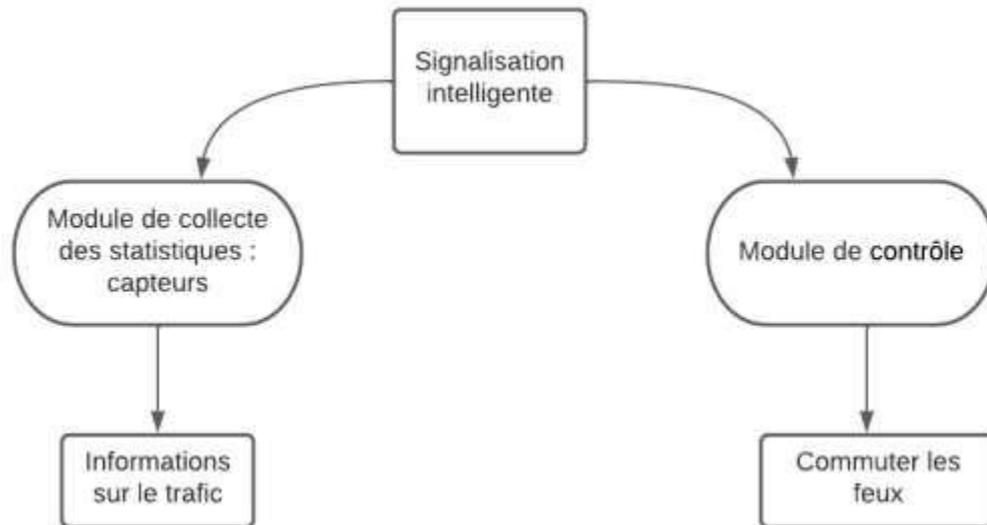


Figure 10. Structure de la signalisation intelligente.

4.3. Défis dans la conception de la signalisation intelligente

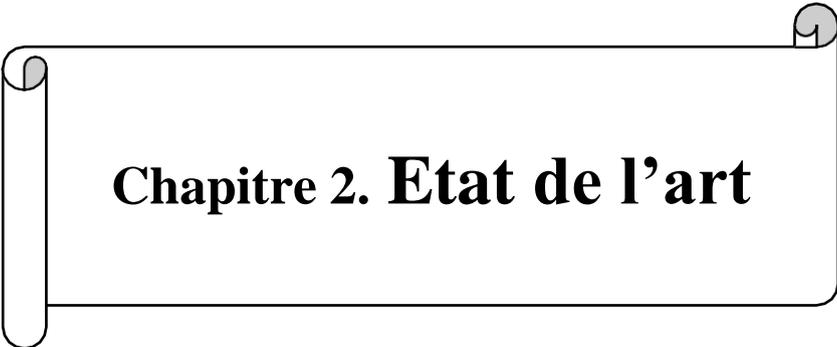
Aujourd'hui, le nombre de véhicule dans le monde a terriblement augmenté. Cependant, l'infrastructure des routes reste la même ce qui résulte en des problèmes complexes pour la signalisation existante. La signalisation intelligente reste la solution unique pour résoudre ce problème et alléger la congestion au niveau des routes. Nous pouvons souligner plusieurs avantages de la signalisation intelligente et qui reste un objectif vivement souhaité par les citoyens et le gouvernement.

4.4. Avantages de la signalisation intelligente [11]

- Fluidifier la route en diminuant les embouteillages.
- Eliminer la conduite "Stop-Start" qui est inefficace.
- Prioriser la circulation en réponse à la demande en temps réel.
- Réduire la pollution
- Réduire le stress.

5. Conclusion

Au sein de ce chapitre, on a défini les Systèmes du Trafic Intelligents (STI) et la signalisation intelligente. On a exposé les avantages et les défis de la signalisation intelligente. Dans le prochain chapitre, on s'intéressera aux travaux reliés sur la signalisation intelligente.



Chapitre 2. Etat de l'art

1. Introduction

Chaque année le nombre des usagers routiers augmente et dépasse la capacité de certaines routes conduisant à des embouteillages et à des files d'attente longues. Ce phénomène de congestion mène à la création des nouvelles techniques de contrôle du trafic parmi eux les feux de signalisation intelligentes.

Dans ce chapitre on va faire une synthèse des travaux reliés à la signalisation intelligente.

2. Travaux reliés

Les travaux existants peuvent être classés selon deux axes : les méthodes mathématiques et les méthodes issues de l'intelligence artificielle.

2.1. Modèles mathématiques

La méthode mathématique la plus utilisée est la Programmation Dynamique (PD). Elle décompose le problème complexe en un ensemble de sous problèmes avec des pas du temps discrets entre eux. Cette méthode utilise souvent des algorithmes de recherche pour aboutir à la solution globale optimale. Plusieurs systèmes de contrôle de feux de circulation adaptatifs ont été basés uniquement sur la méthode PD [14,15].

Les auteurs dans [16] ont aussi proposé un algorithme intelligent de contrôle de feux de circulation, basé sur la PD mais combinée avec les réseaux de neurones. En supposant que le futur schéma d'arrivée des véhicules est connu, PD est utilisé pour trouver les temps des feux verts optimaux pour une intersection isolée.

Dans [17], les auteurs ont proposé de nouvelles approches mathématiques pour gérer le système de trafic urbain sur une intersection isolée. En effet, ils ont proposé des approches basées sur la méthode "Branch and Bound" [18] pour le contrôle de feux de circulation. Ils ont aussi utilisé des nouvelles technologies pour traiter le mouvement de chaque véhicule individuellement et distinguer les différents types de véhicules tels que les véhicules publics et d'urgence.

2.2. Modèles issus de l'intelligence artificielle

Pour les solutions issues de l'intelligence artificielle, on retrouve des travaux se basant sur la logique, les méthodes bio-inspirées et les méthodes issues de l'apprentissage par renforcement. Les derniers travaux ont été synthétisés comme suit :

- La logique floue [19] permet de remplacer le choix strictement binaire par la mise en place des degrés de vérification d'une condition. Plusieurs auteurs l'ont utilisé pour le traitement des problèmes de gestion des feux de signalisation [20-21].
- Dans [22], les auteurs ont proposé aussi un feu de signalisation intelligent se basant sur la logique floue. La collecte des données de trafic est effectuée en utilisant un réseau de capteurs sans fil. Ces données collectées forment deux entrées pour le contrôleur logique (quantité de trafic et le temps d'attente pour chaque voie). Puis, ce contrôleur calcule un degré de priorité afin d'attribuer le feu vert pour la valeur la plus élevée.
- Les auteurs dans [23] ont aussi utilisé la méthode Mamdani de la logique floue pour développer une meilleure conception dynamique de contrôle de feux de circulation. La durée du feu vert est prédite en se basant sur trois critères qui sont : le nombre de véhicules, la longueur de la file d'attente et la largeur de chaque voie où chacun de ces critères prend une valeur dans des intervalles identifiés précédemment. En utilisant ainsi certaines règles et avec les valeurs des trois critères le calcul de la durée de feu vert est calculée.
- Dans [24], ils ont prouvé qu'ils peuvent améliorer efficacement les performances du contrôleur basé sur la logique floue, en appliquant une méthode bio-inspirée qui est l'Optimisation par Essaim Particulaire (PSO) pour affiner les paramètres du contrôleur de feux de signalisation.
- Le réseau de neurone [19] est inspiré du système nerveux biologique. Il est composé de plusieurs neurones connectés entre eux. Dans cette méthode, l'apprentissage est fait de sorte que pour chaque entrée particulière, il y a une sortie cible en ajustant les poids entre les neurones. Cet ajustement est fait en comparant la sortie du réseau par rapport à la valeur cible attendue, jusqu'à aboutir à la sortie correspondante. Il peut apporter des solutions simples à des problèmes complexes.

- Cet outil a été utilisé aussi pour la conception des feux adaptatifs [25]. En effet, le modèle de la logique floue proposé est combiné à un réseau de neurone pour optimiser le contrôleur du trafic.
- L'algorithme génétique [19] est parmi aussi les outils utilisés pour traiter le problème de gestion du trafic routier [26,27]. A l'aide des informations sur le trafic, tels que le nombre de véhicules et le temps moyen d'attente sur une intersection, cet algorithme peut appliquer des méthodes d'optimisation afin d'aboutir à une solution approchée.
- Dans [5], les auteurs ont proposé d'optimiser le temps pour quatre intersections. Au début ils ont créé le chromosome initial de 16 valeurs aléatoires où chaque gène présente une durée de feu vert et à partir de cette représentation, la population initiale est générée en permutant ses gènes. Puis des opérateurs de mutation et croisement sont appliqués aux individus. Ils ont utilisé une sélection par roulette et ce processus se répète jusqu'à aboutir à un nombre de génération bien défini.
- Comme l'Apprentissage par Renforcement (AR) est une méthode d'apprentissage dynamique, il a été vivement adopté par plusieurs travaux de conception de feux de circulation intelligente. Ce type d'apprentissage aide le système à effectuer une série d'actions optimales sur l'environnement pour obtenir une récompense instantanée et maximiser ainsi le bénéfice cumulé au fil du temps. Toutefois, il est nécessaire de définir l'ensemble d'états, l'ensemble d'action et les fonctions de récompense.

Plusieurs recherches ont adopté l'apprentissage par renforcement pour la conception des modèles de trafic intelligents. Cependant, les premiers travaux étaient limités par de simples simulations et un manque de puissance de calcul [28, 29]. Ces travaux ont été ensuite améliorés pour créer une variété d'outils de simulation de plus en plus complexes et réalistes.

- Les travaux d'Araghi et al [30] et Prabuchandrn et al [31] ont utilisé l'apprentissage par renforcement pour gérer les feux de circulation. Dans [32] ils ont construit deux types d'algorithmes d'apprentissage par renforcement, le premier utilise la politique du gradient profond et le deuxième algorithme basé sur la fonction de valeur qui permet de prédire le meilleur signal du trafic dans les intersections. Le système de contrôle adaptatif des feux reçoit l'état actuel et produit un signal de contrôle adaptatif.

Contrairement aux travaux basés sur la politique du gradient profond qui mappent leurs observations directement, les agents basés sur la fonction de valeur estiment les valeurs pour tous les signaux de contrôle légaux, ensuite ils sélectionnent l'action de contrôle optimale avec la valeur la plus élevée.

- Dans [33], ils ont conçu un contrôleur intelligent de feux de circulation. Ils ont utilisé l'algorithme d'apprentissage Q-learnig en tenant compte des informations du trafic aux intersections voisines aussi. Ces informations sont captées à l'aide des caméras.
- Dans [34], les auteurs ont conçu un contrôleur de feux de circulation à l'aide d'un algorithme d'apprentissage Q-asynchrone à n étapes sur deux intersections. L'agent a été présenté par un réseau de neurones à deux couches cachées. L'état de l'environnement est basé sur des statistiques du trafic et des informations sur la phase actuelle de feux de signalisation. Les actions ont été modélisées par l'activation des phases vertes. La récompense est définie en fonction du nombre de véhicules en file d'attente. Les auteurs ont développé une simulation dynamique et stochastique à des heures de pointe pour tester les performances de l'agent.

Tous ces travaux ont été synthétisés dans le tableau comparatif suivant tout en soulignant les avantages et les problèmes de chaque méthode utilisée.

Méthodes utilisées	Auteurs et année	Références	Avantages	Problèmes
Programmation dynamique	T. H. Heung et al, 2005 J. Wu et al, 2009 D. Teodorović et al, 2006	[14] [15] [16]	Elle Fournit une solution globalement optimale. Elle est très flexible, en permettant l'ajout des fonctions ou bien des contraintes quelconques.	Méthode très couteuse en espace mémoire (calculs intermédiaires)
Branch and bound	YAN et al, 2008	[17]	La complexité temporelle est moindre	Il est limité aux réseaux de petite taille.

Logique floue	Z. Liao et L. Zhao, 2006 R Patil et al, 2016 R Hawi et al, 2017 D. Hartanti et al, 2019	[20] [21] [22] [23]	Il n'est pas nécessaire de faire un modèle mathématique du système à régler.	Précision de réglage en général peu élevé. Exigences de calcul très élevées pour un haut degré de précision.
Optimisation par essaim particulière	Y. Wei et al : 2008	[24]	Convergence rapide.	Stagne dans des optima locaux. Le temps de calcul est long pour les instances de grande taille.
Algorithme génétique	X.F. Chen et Z.-K. Shi, 2002 K. B. Kesur , 2009 Y.Aibeche , 2020	[26] [27] [5]	Converge vers la solution optimale.	Le temps de calcul est trop long. Paramètres difficiles à initialiser : fonction d'évaluation, codage de données.
Réseau de neurones	E. Bingham et al , 2001	[25]	Flexible et adaptatif.	Difficulté de déterminer les paramètres du réseau utilisé. Difficulté de

Apprentissage par renforcement	T. L. Thorpe et C. W. Anderson, 1996	[28]	S'adapte bien à la dynamique de l'environnement	Il exige beaucoup de temps et de puissance de calcul pour entraîner le modèle.
	M. Wiering et al : 2000	[29]		
	S. Araghi et al : 2013	[30]		
	K. Prabuchandran , 2014	[31]		
	S. S. Mousavi et al , 2017	[32]		
	Y.Liu et al, 2017	[33]		
	W. Gender , 2018	[34]		

Tableau 1. Synthèse des travaux sur la signalisation intelligente.

Parmi ces méthodes, on s'intéresse à l'apprentissage par renforcement grâce à sa dynamique. Dans le prochain tableau on va synthétiser les travaux précédemment définis :

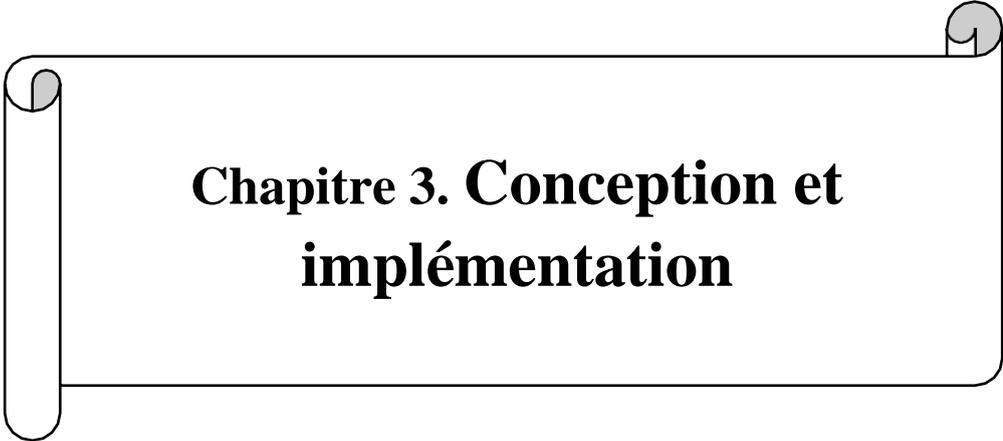
Référence	Algorithme utilisé	Nombre d'intersections	Récompense
[28]	SARSA	Une intersection.	/
[29]	Q-Learning et TD Learning	6 intersections.	Temps d'attente cumulé.
[30]	Multi-agent Q-Learning	50 intersections.	Longueur moyenne de la file d'attente.
[31]	Q-Learning	9 intersections et 20 intersections.	/
[32]	Politique du gradient profond et	Une intersection.	Temps d'attente cumulé.

	apprentissage par renforcement basé sur la fonction de valeur		
[33]	Multi-agent Q-Learning	33 intersections.	Longueur moyenne totale de la file d'attente.
[34]	Q-Learning asynchrone à n étapes.	Une intersection.	Nombre de véhicules dans la file d'attente.

Tableau 2. Synthèse des travaux sur l'apprentissage par renforcement.

3. Conclusion

Vu la dynamique du trafic et les travaux reliés présentés ci-dessus, on peut souligner l'importance de l'apprentissage par renforcement pour la conception des systèmes intelligents de contrôle du trafic. Dans ce cadre, on se propose d'utiliser cette méthode d'apprentissage pour la conception de notre modèle et qui sera détaillée dans le chapitre suivant.



Chapitre 3. Conception et implémentation

1. Introduction

Les méthodes d'apprentissage par renforcement sont des outils adaptés au modèle dynamique du trafic. C'est dans ce cadre que s'insère notre modèle proposé. Ce type d'apprentissage se base sur le modèle décisionnel de Markov : Un outil mathématique puissant de modélisation de la dynamique de l'environnement.

Dans ce qui suit, on présente dans la première partie de ce chapitre les concepts clés de cette modélisation. Elle sera suivie par la présentation du modèle proposé impliquant l'algorithme d'apprentissage par renforcement choisi (Q-learning) et amélioré par une autre technique d'apprentissage (Deep-Learning). Les résultats de l'implémentation seront exposés à la fin de ce chapitre.

I. Modélisation et conception

1. Concepts clés de la modélisation

1.1. Processus de Décision de Markov (PDM)

C'est un outil de formalisation de la prise de décision séquentielle. C'est à dire, pour un processus qui passe par un ensemble d'états, s'il est à un instant t dans l'état s_t , alors la distribution de la probabilité conditionnelle de l'état suivante s_{t+1} ne dépend que de l'état qui la précède [35]. Le modèle de Markov forme un moyen très utile pour la description du comportement dynamique d'un agent [36]. Donc, il représente la base de la structuration des problèmes résolus par l'apprentissage par renforcement [35].

1.2. Apprentissage par renforcement

L'apprentissage par renforcement est un type d'apprentissage automatique qui permet à un agent d'apprendre quelle action à exécuter à partir d'une certaine situation [35]. L'agent n'est pas informé des actions à entreprendre. Donc, il explore son environnement dans le but de maximiser une récompense reçue de son l'environnement [36]. Il cherche non seulement à maximiser la récompense actuelle, mais plutôt ses récompenses futures aussi [37].

Comme on a dit précédemment, la structure d'apprentissage par renforcement peut être décrite comme un PDM, composé de :

- **L'agent** : c'est l'unité fondamentale de l'apprentissage par renforcement. Il interagit séquentiellement avec son environnement, en disposant de l'intelligence essentielle pour prendre des décisions et choisir la bonne action pour une situation donnée [35].
- **L'environnement** : Il représente le monde avec lequel l'agent interagit et apprend [38].
- **L'ensemble d'états** : représente toutes les représentations possibles de l'environnement [38].
- **L'ensemble d'actions** : représente tous les actions possibles que l'agent peut effectuer [38].
- **La récompense** : le résultat d'exécuter une action a_t à partir d'une situation s_t .

La structure du modèle d'apprentissage par renforcement est représentée dans la figure suivante :

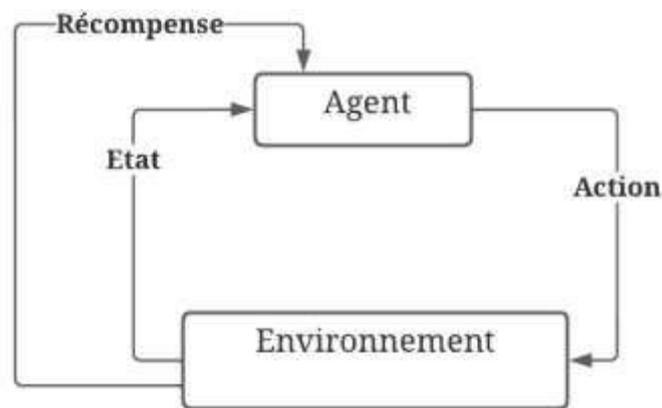


Figure 11. Structure d'apprentissage par renforcement.

Plusieurs algorithmes d'apprentissage par renforcement existent dans la littérature, le plus utilisé est le Q-Learning.

a. Q-learning

Q-learning est l'un des algorithmes fondamentaux d'apprentissage par renforcement [38]. IL permet de stocker toutes les valeurs Q d'état-action possibles pour une politique optimale de l'agent dans un tableau. La mise à jour de ce tableau est établie en utilisant l'équation (1) [39] :

$$Q(s, a) = Q(s, a) - \alpha \cdot (Q(s, a) - r - \gamma Q(s', a'^*)) \quad (1)$$

Pour une convergence plus rapide [40], cette équation a été remplacée par (2) :

$$Q(s_t, a_t) = r_{t+1} + \gamma \cdot \max_A Q'(s_{t+1}, a_{t+1}) \quad (2)$$

Où :

- r_{t+1} signifie la récompense reçue après avoir exécuter l'action a_t dans l'état s_t .
- $Q'(s_{t+1}, a_{t+1})$ est la valeur Q associée à l'action a_{t+1} dans l'état s_{t+1} , c'est-à-dire l'état suivant après l'action a_t dans l'état s_t .
- γ est un facteur d'actualisation qui aide à dénoter une légère pénalisation de la récompense future par rapport à la récompense immédiate.

b. Politique décisionnelle

L'agent suit une politique décisionnelle selon les valeurs optimales de Q . On la désigne par le symbole π . C'est une fonction qui met en correspondance un état donné avec les probabilités de sélectionner chaque action possible dans cet état [35]. Néanmoins, la politique peut changer pour avoir une meilleure exploration des solutions en utilisant d'autres politiques existantes (ϵ -greedy, softmax,...).

D'autre part, en utilisant les algorithmes d'apprentissage par renforcement, l'espace d'état est généralement grand qu'il est impossible de découvrir et de sauvegarder chaque paire état-action (Valeurs Q). Par conséquent, on propose dans ce travail d'approcher ces valeurs avec un modèle d'apprentissage supervisé dans notre cas l'apprentissage profond (Deep-Learning).

1.3. Apprentissage profond

Récemment, l'apprentissage profond est mis en œuvre grâce à l'apparition des données massives et la puissance des ordinateurs [41]. Donc, l'apprentissage profond ou bien le Deep Learning (communément appelé) est un apprentissage réalisé avec un réseau de neurones avec plusieurs couches cachées. Il repose sur un apprentissage hiérarchique couche par couche ou la sortie d'une première couche sera l'entrée de la deuxième et ainsi de suite [42]. Le Deep-learning combiné à l'algorithme du Q-learning ont été utilisés dans notre modèle proposé pour résoudre le problème de signalisation intelligente. Il est présenté ci-dessous.

2. Modèle Proposé

Pour adapter le modèle proposé à notre problématique (signalisation intelligente), on doit modéliser tout d'abord les composants de l'algorithme Q-learning et qui sont : l'agent, l'environnement, les états, les actions et la récompense.

2.1. Modélisation

a. L'agent

L'agent représente le système de contrôle des feux de signalisation qui interagit avec l'environnement dynamique.

b. Environnement

Représentant le trafic à l'intérieur d'une intersection.

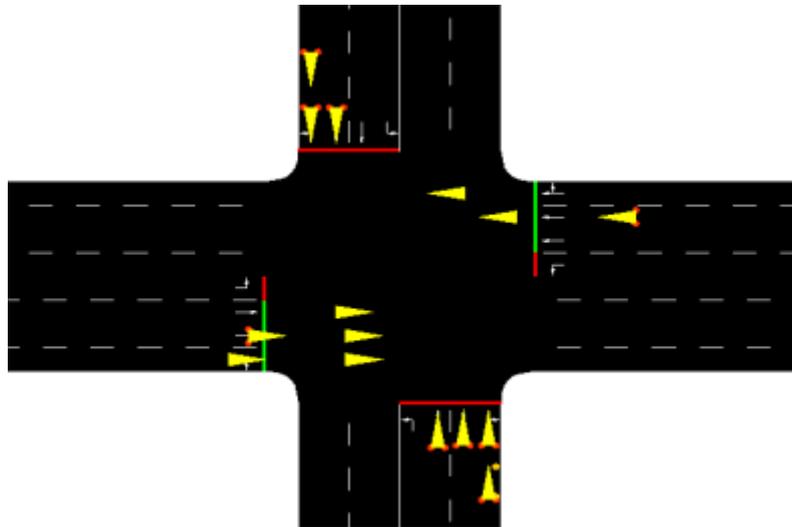


Figure 12. Environnement.

c. L'état perçu par l'agent

L'état forme une représentation de l'environnement à un instant t noté s_t . Il est modélisé dans notre modèle par les informations spatiales des véhicules hébergés sur le réseau, sous forme d'un vecteur détaillant la présence ou non d'un véhicule dans l'espace routier.

d. L'action

Une action de l'agent représente l'activation d'une phase verte pour un ensemble de voies à une durée fixée. On a quatre actions possibles définies comme suit :

- Vague Nord-Sud : la phase verte est active pour les véhicules qui se trouvent dans les tronçons Nord et Sud et qui veulent continuer tout droit ou tourner à droite.

- Vague Nord-Sud Gauche : la phase verte est active pour les véhicules qui se trouvent dans les tronçons Nord et Sud et qui veulent tourner à gauche.

- Vague Est-Ouest : la phase verte est active pour les véhicules qui se trouvent dans le tronçon Est et Ouest et qui veulent continuer tout droit ou tourner à droite.

- Vague Est-Ouest à gauche : la phase verte est active pour les véhicules qui se trouvent dans le tronçon Est et Ouest et qui veulent tourner à gauche.

On peut représenter ces actions par les figures suivantes :

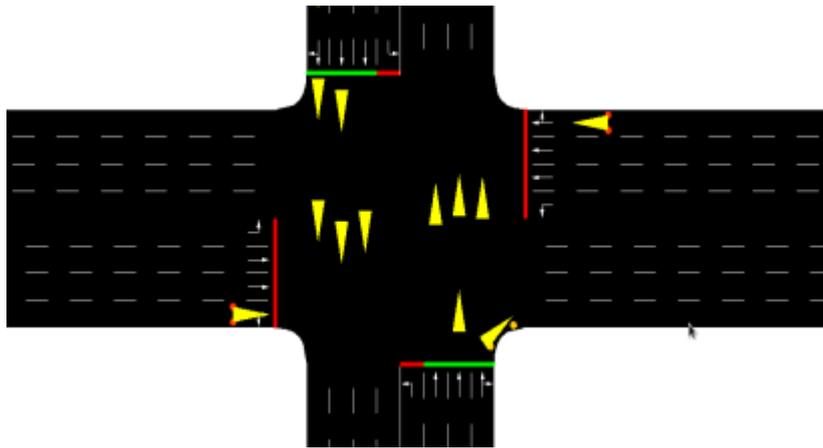


Figure 13. Vague Nord-Sud.

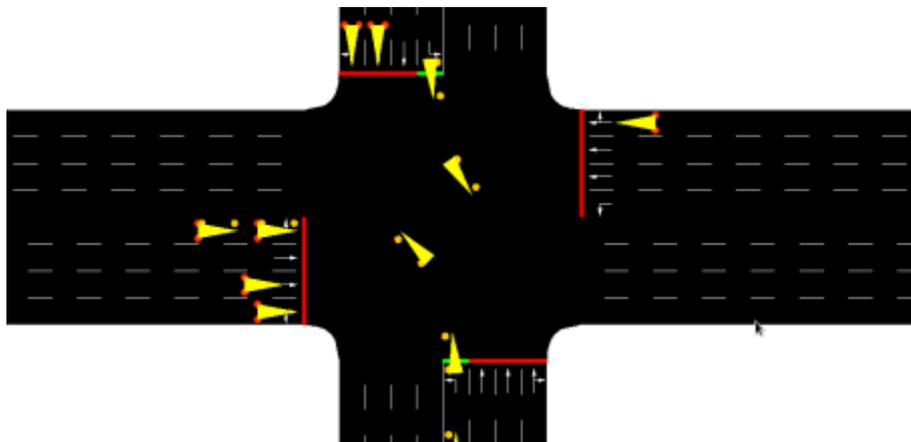


Figure 14. Vague Nord-Sud gauche.

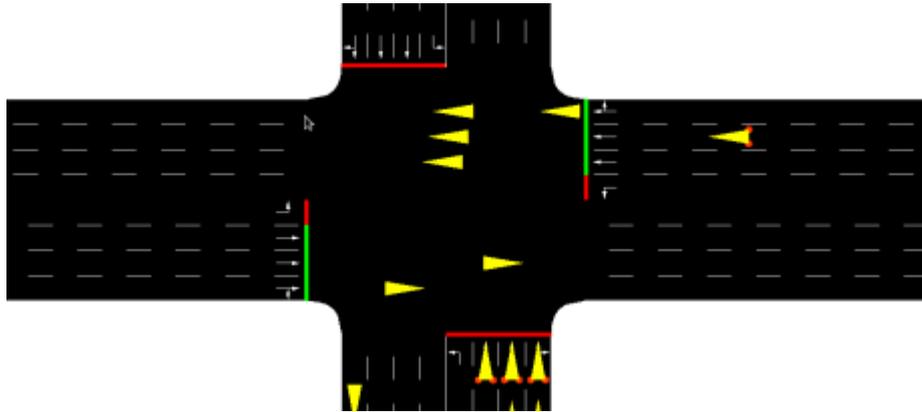


Figure 15. Vague Est-West.

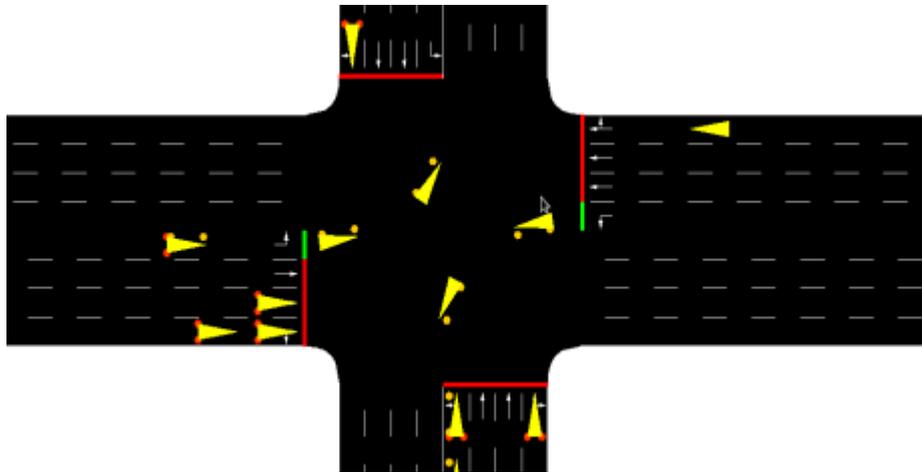


Figure 16. Vague Est-West gauche.

e. La récompense

Comme on a vu précédemment, la récompense est le résultat que l'environnement donne à l'agent après avoir exécuter une action a_t à partir d'un état s_t . Dans cette application, l'objectif est de fluidifier le flux de trafic à travers le carrefour dans le temps. La récompense doit être dérivée donc d'une mesure de performance de l'efficacité du trafic. Plusieurs mesures ont été utilisées dans la littérature telle que le débit, le retard moyen, le temps d'attente et le temps de parcours. Dans notre modèle, on a choisi d'utiliser le temps d'attente cumulé jugé comme un paramètre important pour la fluidification du réseau routier. Il est calculé par l'équation (3)

$$TAC_t = \sum_{veh=1}^n TA(veh, t) \quad (3)$$

Où :

- TAC_t est le temps d'attente total cumulé à l'instant t .
- $TA_{t(veh,t)}$ est la durée en secondes pendant laquelle un véhicule veh est en attente à l'instant t , depuis son apparition dans l'environnement.
- n est le nombre total de véhicules dans l'environnement à l'instant t .

La 2ieme étape de notre modèle proposé est l'amélioration du Q-learning. Cette dernière est représentée ci-dessous

2.2. Modèle du Q-learning amélioré par le Deep-learning

L'agent doit choisir la meilleure action qui maximise la fonction $Q(s_t, a_t)$ présenté dans l'équation (2). Pour améliorer l'apprentissage, on a utilisé un réseau de neurones profond entièrement connecté. Ce réseau de neurone profond est composé de :

- Une Couche d'entrée représentant les valeurs Q (états actions) dépendant de la modélisation de l'intersection.
- Un nombre de Couches cachées utilisées pour l'apprentissage (plus le nombre de couches augmente plus l'apprentissage est mieux mais le temps d'exécution augmente)
- la couche de sortie avec 4 neurones, chacune représentant la valeur d'une action selon un état existant.

Les informations nécessaire à l'apprentissage sont fournis à l'agent sous forme d'un échantillon, ou chaque échantillon contient l'état actuelle s_t , l'action choisie a_t , la récompense reçue r_t et le prochain état s_{t+1} . A partir de ces informations extraites des échantillons, l'entraînement cherche à approcher la fonction Q . Chaque échantillon passe par un ensemble d'opérations :

- D'abord, la prédiction des valeurs Q du $Q(s_t)$, qui représente la connaissance actuelle de l'agent sur les valeurs d'action de l'état s_t .
- Après, la prédiction des valeurs Q de $Q'(s_{t+1})$ qui sont la connaissance de l'agent sur les valeurs d'action à partir de l'état s_{t+1} .
- Puis, la mise à jour de la valeur $Q(s_t, a_t)$ à l'aide de l'équation(2).

Les valeurs $Q(s_t, a_t)$ incluent maintenant la récompense future maximale attendue grâce à la mise à jour de la valeur Q .

Une fois que le réseau neuronal profond s'est suffisamment approché de la fonction Q-learning, il reste juste de choisir la politique de sélection d'une action.

-Politique du choix d'action

Pour notre modèle, on a choisi d'utiliser la politique ϵ -greedy pour obtenir un équilibre entre l'exploration et l'exploitation. Ce ϵ est la probabilité que notre agent explore son environnement. Il est fixé au départ à 1 pour assurer que l'agent explore son environnement. Puis, cette valeur est dégradée pour que l'agent exploite ce qu'il a appris [35].

Après l'explication des différentes parties de notre modèle, on peut le présenter dans la figure suivante :

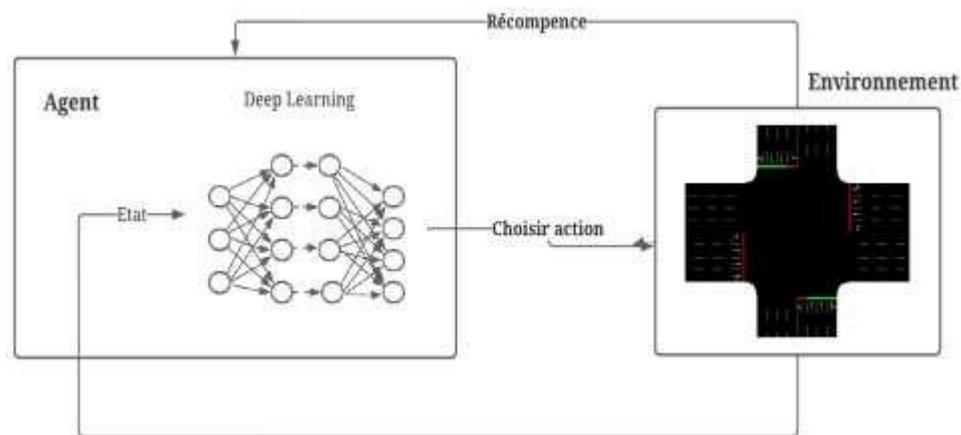


Figure 17. Modèle proposé.

3. Modèle de simulation proposé

Pour valider le modèle proposé, il fallait proposer aussi une autre modélisation d'intersection sur la plateforme SUMO (Simulation Of Urban Mobility) qui fournit une interface de programmation d'application (API). A l'aide de cette interface, on a créé une intersection à quatre tronçons modélisant l'environnement de l'agent défini précédemment (figure 12). Chaque tronçon est composé de quatre voies entrantes et quatre voies sortantes.

Chaque voie entrante présente les directions qu'une voiture peut suivre. La voie à droite est dédiée au virage à droite ou continuer tout droit. Les deux voies médianes sont dédiées pour aller tout droit uniquement. Ces trois voies partagent le même feu de signalisation. La voie à gauche est dédiée seulement au virage à gauche. Un feu de signalisation est dédié pour cette voie.

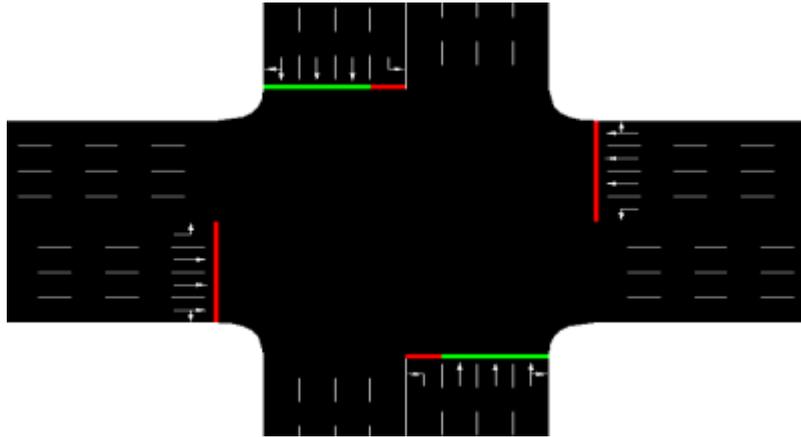


Figure 18. Intersection modélisée.

Afin de fournir à l’agent une représentation de l’environnement les voies entrantes sont discrétisées en cellules permettant d’identifier la présence ou l’absence d’un véhicule à l’intérieur. Chaque tronçon est divisé en cellules. La moitié d’entre eux sont situées le long de la voie de gauche et les autres couvrent les trois autres voies.

Les véhicules sont représentés par des triangles jaunes (qui représentent les entités mobiles), Les feux de signalisation sont situés au centre de l’intersection et sont contrôlés par l’agent. Si l’agent à l’instant $t+1$ choisit la même action de l’instant t , une phase jaune est activée. Sinon, il n’y a pas de phase jaune. Cette modélisation a été proposée dans un fichier `environnement.xml`. Pour valoriser le modèle proposé, on a utilisé une distribution aléatoire pour générer les véhicules. A l’aide d’un générateur de nombres aléatoires, le tronçon source et la destination de chaque véhicule sont déterminés.

II. Implémentation et expérimentation

1. Matériel utilisés pour le développement de l’application

Nous avons utilisé le matériel suivant :

Processeur	Intel (R) Core i5-4300M CPU @ 2.60GHZ x4
Système d’exploitation	Linus Mint 19.2 Tina 64 bits
Disque dur	250 GB
RAM	4 GB

Tableau 3. Matériel utilisé.

2. Outils logiciels utilisés pour le développement de l'application

✦ **Langage Python 3.6.9** : Python est un langage de programmation de haut niveau développé par Guido Van Rossum et de nombreux contributeurs bénévoles. Il est un langage portable, dynamique, extensible, gratuit qui permet une approche modulaire et orienté objet de la programmation.

Il dispose d'une bibliothèque standard vaste et complète.

✦ **SUMO 1.1.0** : est un package de simulation de trafic open source conçu pour gérer de grands réseaux. Il permet une simulation intermodal incluant les piétons et fournis un large éventail d'outils pour la création des scénarios.

✦ **Tensorflow** : C'est une plate-forme open source pour l'apprentissage automatique. Elle fournit un écosystème complet et flexible d'outils, de bibliothèques et de ressources communautaires qui permettent aux développeurs de créer et développer facilement des applications de "machine learning".

✦ On a aussi utilisé des fonctions des bibliothèques Traci et Numpy.

3. Exécution et résultats

Pour le choix des paramètres utilisés dans l'application, on a pris en considération le temps d'exécution et la qualité de l'apprentissage et de la solution obtenue.

3.1. Paramètres du Q-Learning

Paramètre	Valeur
Gamma : γ	0.75

Tableau 4. Paramètre du Q-Learning.

3.2. Paramètres du Deep-Learning

Paramètres	Valeurs
Nombre de couches cachées	4
Nombre de neurones d'une couche cachée	200

Nombre d'états	80
Nombre d'actions	4
Taux d'apprentissage	0.001

Tableau 5. Paramètres du Deep-Learning.

3.3. Tableau d'initialisation des paramètres

Paramètres	Valeurs
Nombre d'épisodes	100
Nombre d'étapes par épisode	5400
Nombre de véhicules	1000
Durée de la phase verte	12
Durée de la phase jaune	4

Tableau 6. Paramètres d'initialisation.

4. Déroulement de l'exécution

L'exécution du programme est faite sur le Terminal. Une phase d'entraînement est lancée, suivie d'une phase de simulation présentée par les figures suivantes :

```

oem@oem: ~/Desktop/PFE/APP1
File Edit View Search Terminal Help
(base) oem@oem:~$ cd /home/oem/Desktop/PFE/APP1
(base) oem@oem:~/Desktop/PFE/APP1$ python Entraînement.py

```

Figure 19. Lancement de l'entraînement.

```
oem@oem: ~/Desktop/PFE/APP1
File Edit View Search Terminal Help
Retrying in 1 seconds
Loading configuration... done.
*La récompense : -32044.0 *La valeur d epsilon: 0.88
*Le temps de simulation est : 4.7 s et le temps d entraînement est : 16.3 s donc
le temps total est : 21.0 s

** Episode numéro : 14 parmi 100 épisodes.
Retrying in 1 seconds
Loading configuration... done.
*La récompense : -30471.0 *La valeur d epsilon: 0.87
*Le temps de simulation est : 4.5 s et le temps d entraînement est : 16.4 s donc
le temps total est : 20.9 s

** Episode numéro : 15 parmi 100 épisodes.
Retrying in 1 seconds
Loading configuration... done.
*La récompense : -30209.0 *La valeur d epsilon: 0.86
*Le temps de simulation est : 4.6 s et le temps d entraînement est : 16.0 s donc
le temps total est : 20.6 s

** Episode numéro : 16 parmi 100 épisodes.
Retrying in 1 seconds
Loading configuration... done.
*La récompense : -22267.0 *La valeur d epsilon: 0.85
```

Figure 20.Modèle en cours d'entraînement.

```
oem@oem: ~/Desktop/PFE/APP1
File Edit View Search Terminal Help
** Episode numéro : 98 parmi 100 épisodes.
Retrying in 1 seconds
Loading configuration... done.
*La récompense : -8797.0 *La valeur d epsilon: 0.03
*Le temps de simulation est : 4.5 s et le temps d entraînement est : 16.4 s donc
le temps total est : 20.9 s

** Episode numéro : 99 parmi 100 épisodes.
Retrying in 1 seconds
Loading configuration... done.
*La récompense : -7974.0 *La valeur d epsilon: 0.02
*Le temps de simulation est : 4.5 s et le temps d entraînement est : 16.6 s donc
le temps total est : 21.1 s

** Episode numéro : 100 parmi 100 épisodes.
Retrying in 1 seconds
Loading configuration... done.
*La récompense : -8235.0 *La valeur d epsilon: 0.01
*Le temps de simulation est : 4.5 s et le temps d entraînement est : 16.5 s donc
le temps total est : 21.0 s

** Temps du début : 2022-06-28 18:29:39.988104
** Temps de la fin: 2022-06-28 19:08:49.530129
(base) oem@oem:~/Desktop/PFE/APP1$ python test.py
```

Figure 21. Fin d'entraînement.

Maintenant la phase de simulation peut commencer après cette phase d'entraînement.

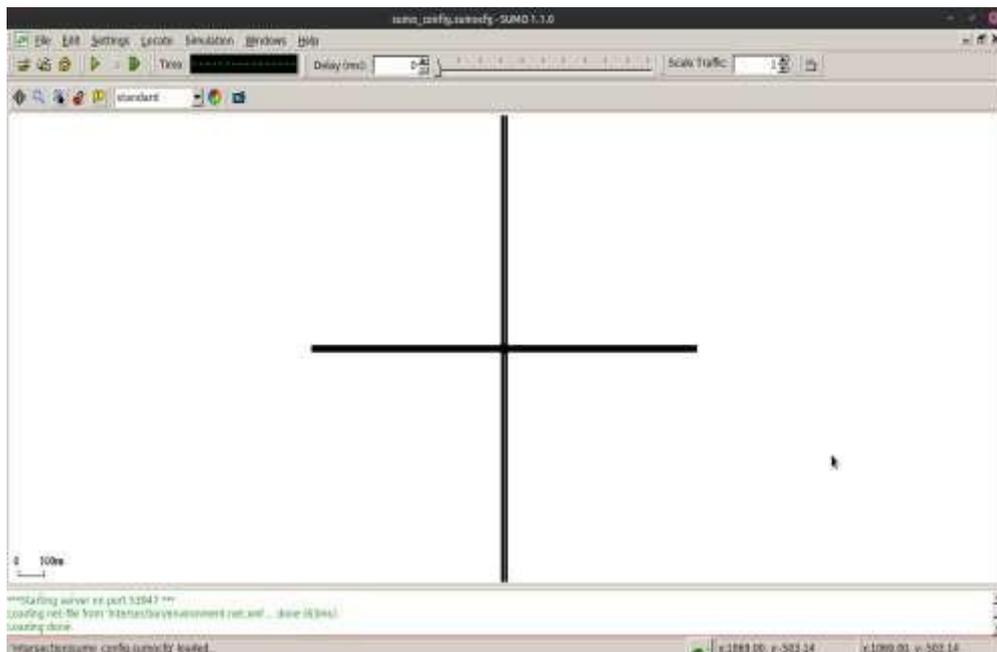


Figure 22. Présentation de l'intersection sur SUMO.

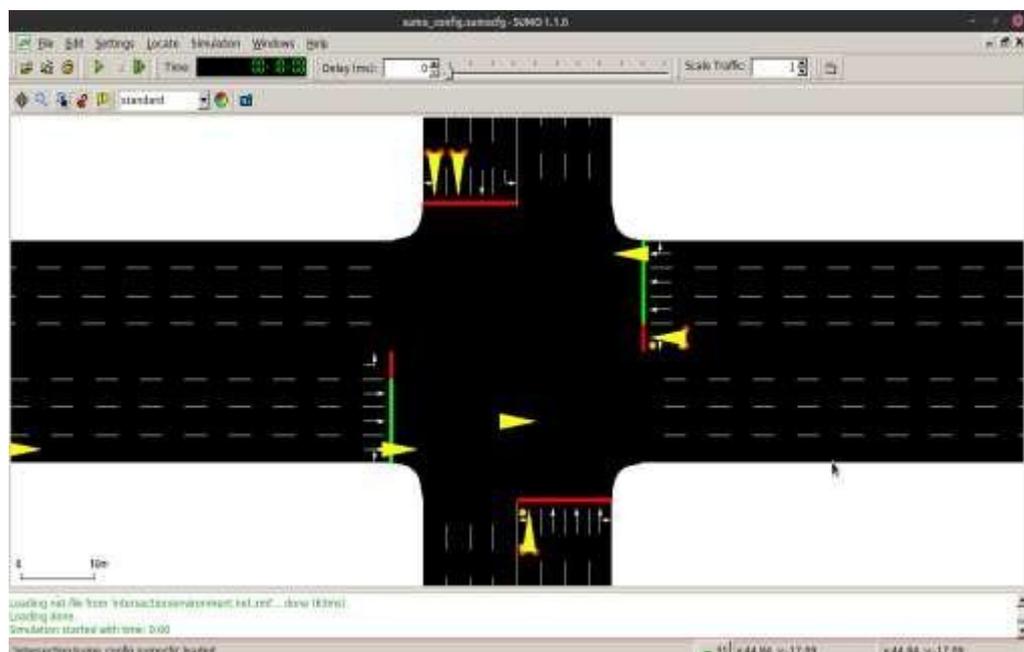


Figure 23. Simulation en cours d'exécution.

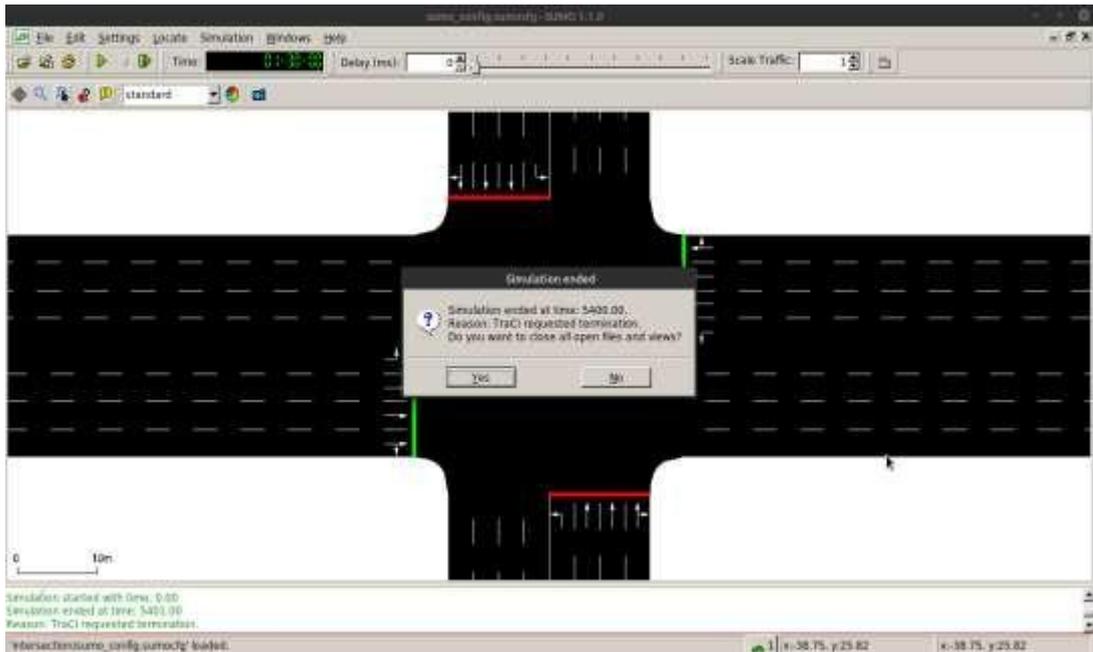


Figure 24. Fin de la simulation.

Les résultats obtenus ont été présentés sous forme de graphes valorisant la performance du système proposé pour les étapes d'entraînement et de test et sont exposés ci-dessous.

5. Discussion des résultats

- **Phase d'Entraînement**

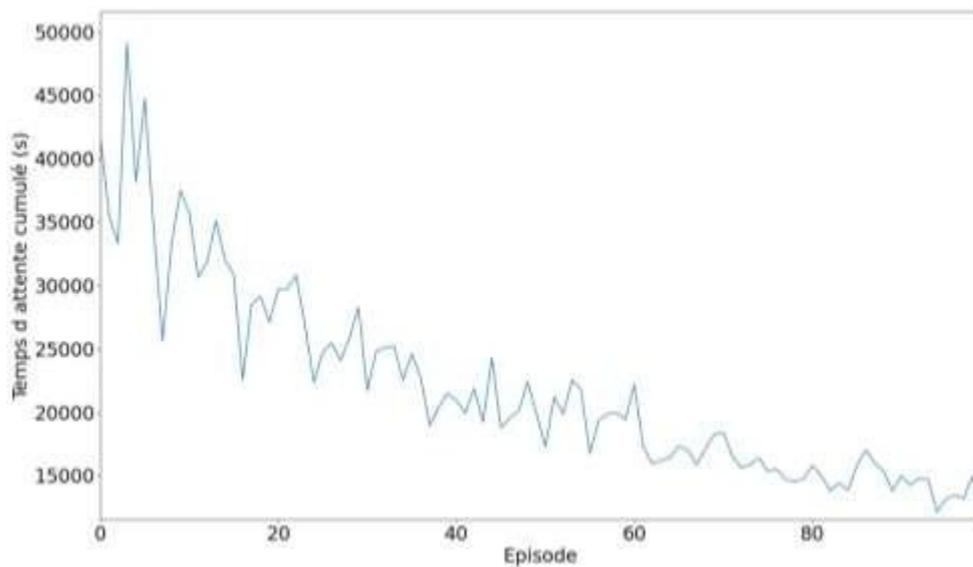


Figure 25. Temps d'attente cumulé pour la phase d'entraînement.

La figure 25 présente le résultat d'entraînement du notre modèle en fonction de temps d'attente cumulé. On remarque que le temps d'attente diminue avec le nombre d'épisodes, ce qui veut dire que notre modèle est bien entraîné.

◦ **Phase de Test** : Les résultats discutés sont le nombre de véhicules (Queue) en fonction du débit du trafic.

Pour un débit faible, le nombre de véhicules entrant est initialisé à 100 véhicules. Pour le moyen débit, le nombre de véhicules varie entre 200 et 500 véhicules par contre pour un haut débit ; le nombre de véhicules dépasse les 800 véhicules.

- **Faible débit**

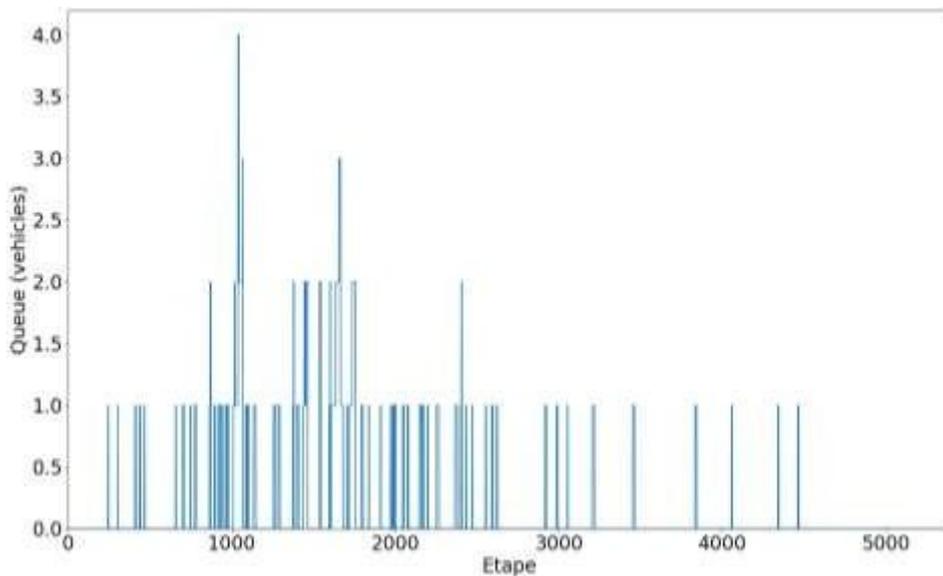


Figure 26. Queue pour un faible débit (100 véhicules).

- **Moyen débit :**

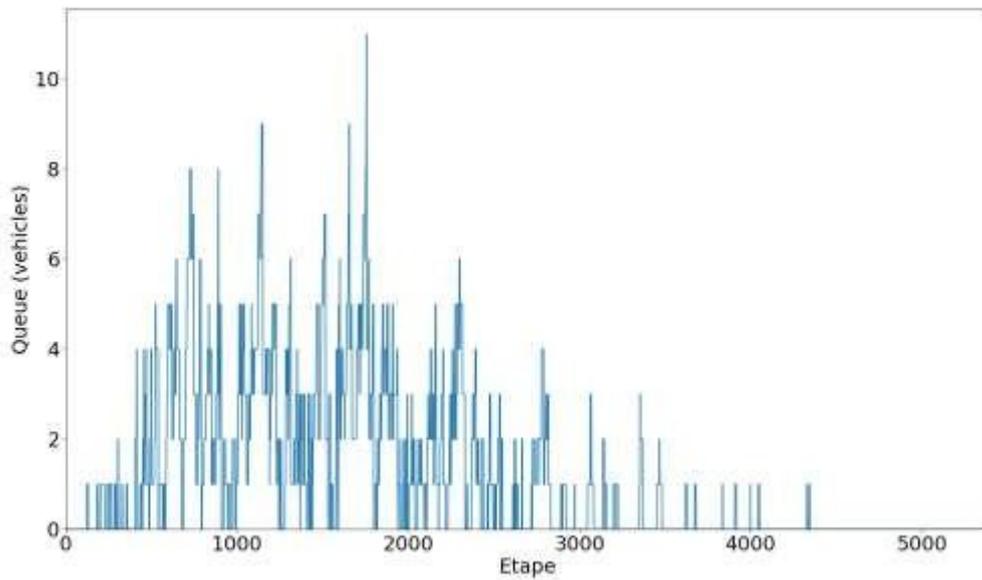


Figure 27 Queue pour un trafic moyen (500 véhicules).

- **Haut débit :**

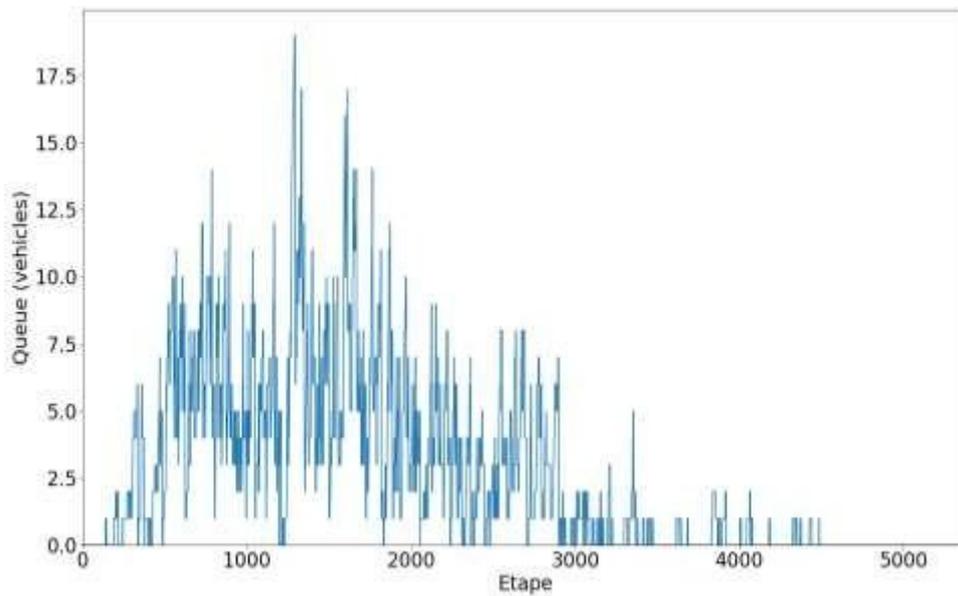


Figure 28. Queue pour un haut débit (1000 véhicules).

Les figures précédentes présentent la file d'attente résultant du test du modèle entraîné. Dans les trois figures, on remarque qu'il y a des pics à certains moments. Ces pics se différencient d'une figure à une autre selon les types de débit.

Pour la figure 26 : 4 véhicules au maximum, la figure 27 : 12 véhicules au maximum et la figure 28 : 18 véhicules au maximum. On remarque aussi que le nombre de véhicules n'augmente pas d'une façon exponentielle, au contraire après chaque pic le nombre de véhicules diminue.

Ces résultats reflètent l'adaptation du modèle proposé aux différents types de trafic et assurent :

- Une adaptation de la signalisation
- Elimination de la congestion.
- Fluidification du trafic.

6. Conclusion

Dans ce chapitre, on a présenté un modèle hybride proposé pour la signalisation intelligente (Q-learning avec Deep-Learning). On a aussi exposé notre modélisation du trafic à l'intérieur de l'intersection à l'aide de la plateforme SUMO. Les résultats obtenus permettent d'appuyer le modèle proposé et de souligner son adaptation au trafic.

Conclusion générale

La congestion routière forme un grand problème pour le développement de notre société. Plusieurs travaux ont été réalisés pour la résolution de ce problème. Le but de notre travail est de proposer une nouvelle solution pour ce problème afin de fluidifier le trafic routier.

Dans ce mémoire, nous avons présenté des notions importantes sur les réseaux routiers et les systèmes intelligents de gestion de ces réseaux dans une première partie. Ensuite, dans la deuxième partie nous avons présenté les travaux connexes sur les contrôles de feux de signalisation intelligente.

Notre recherche nous a amené à proposer une nouvelle solution pour la gestion dynamique de la signalisation intelligente en utilisant une hybridation entre deux algorithmes d'apprentissage qui sont l'apprentissage par renforcement (Q-learning) et le Deep Learning.

Le système proposé a été testé pour différents débits de trafic, où on a constaté que notre système est adaptatif au grand débit de trafic et répond ainsi à l'objectif attendu qui est l'élimination de la congestion aux heures de pointe.

La dernière partie du mémoire a été consacrée à l'implémentation de notre modèle proposé avec le langage Python et sa modélisation sur la plate-forme Sumo qui a permis de mieux appréhender le comportement des feux de signalisation aux différents débits du trafic.

En perspectives ; on vise à améliorer les résultats obtenus, par l'ajout de plusieurs intersections. Aussi, on s'intéressera à l'adaptation des feux de signalisation avec les véhicules prioritaires (police, ambulance).

Références bibliographiques

- [1] I.ALILOUA ,et A.DJOUDER . Gestion de Trafic Urbain à base de Réseau de Capteurs sans Fil : Cas de la Ville de Bejaia . Mémoire de fin de cycle en vue de l'obtention du diplôme de Master en Recherche Opérationnelle, 2012
- [2] S.Hadj Rabah, D.Nouari, Feux Tricolores Intelligents pour les Villes Intelligentes, Mémoire de Master, Université Akli Mohand Oulhadj de Bouira, 2019.
- [3] Guideke B.R, Valerie M.O.M.R. Etude et Simulation des feux de circulation cas du carrefour poste centrale de Yaoundé Caméroun. 2018.
- [4] Jia Wu, Utilisation de la conduite coopérative pour la régulation de trafic dans une intersection, École Doctorale Sciences Pour l'Ingénieur et Microtechniques, 20 Juillet 2011.
- [5] Y.Aibeche, Développement d'un système intelligent de gestion des feux de circulation à Base des algorithmes génétiques, 2020.
- [6] THAI-PHU NGUYEN , Conception et application d'un modèle de l'information routière et ses effets sur le trafic .Thèse de doctorat, universite paris-est .Page 19, Disponible sur : <https://pastel.archives-ouvertes.fr/tel-00626631/document> , 2010.
- [7] A.Kumar , Développement d'un système intelligent de feux de circulation à l'aide Réseau à tolérance de retard. Thèse en exécution partielle pour l'attribution du diplôme de maître de la technologie dans informatique ,université guru gobind singh indraprastha 2010- 2012.
- [8] Carreira P.S, Esnaul G, Poirier W, Dounoun B, Montassier C, Quinquenel A. Système de transport intelligent – capture et traitement de l'information et prise de décision. INSA : 2015.
- [9] Zantalis F, Koulouras S, Kandris D. A Review of Machine Learning and IOT in Smart Transportation. Future Internet 2019,11,94; DOI:10.3390/fi11040094.
- [10] José,M.Costa. Système de transports intelligents. UIT. Canada:2021, 6-11.
- [11] Labsi M.Y, Ouamri S ; Un système de feux tricolores intelligent de gestion de circulation à Ain Témouchent ; Centre universitaire Belhadj Bouchaib ; 2019/2020.
- [12]Mohamed Tlig, Coordination locale et optimisation distribuée du trafic de véhicules autonomes dans un réseau routier, Ecole doctorale IAEM Lorraine, 26 mars 2015.

- [13] Agrawal A, Paulus R. Intelligent traffic light design and control in smart cities : a survey on a techniques and methodologies. *International Journal of Vehicle Information and Communication Systems*, January 2020. DOI : 10.1504/UVICS2020.111456.
- [14] T. H. Heung, T. K. Ho, and Y. F. Fung, “Coordinated road-junction traffic control by dynamic programming,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 6, no. 3, pp. 341–350, 2005.
- [15] J. Wu, A. Abbas-Turki, and A. El Moudni, “Discrete methods for urban intersection traffic controlling,” in *IEEE 69th Vehicular Technology Conference (VTC Spring)*. IEEE, 2009, pp. 1–5.
- [16] D. Teodorović, V. Varadarajan, J. Popović, M. R. Chinnaswamy, and S. Ramaraj, “Dynamic programming-neural network real-time traffic adaptive signal control algorithm,” *Annals of Operations Research*, vol. 143, no. 1, pp. 123–131, 2006.
- [17] F. Yan, M. Dridi, and A. El Moudni, “A scheduling approach for autonomous vehicle sequencing problem at multi-intersections,” *International Journal of Operational Research*, vol. 9, no. 1, pp. 57–68, 2011.
- [18] W. Glankwamdee. *Topics in Branch and Bound on computational grids*. doctoral thesis. Lehigh University. 2008.
- [19] L. Baghli. *Contribution à la commande de la machine asynchrone, utilisation de la logique floue, des réseaux de neurones et des algorithmes génétiques*. Thèse de doctorat. Université Henri Poincaré, Nancy-I. 1999.
- [20] Z. LIAO and L. ZHAO. *Wireless sensor networks help to improve the traffic safety in residential communities*. In *6th International Conference on ITS Telecommunications Proceedings*, pages 973 -978. June 2006.
- [21] R Patil, Srinivasaraghavan, A Srinivasaraghavan. *Smart Traffic Controller using Fuzy Inference System (STCFIS)*. in *2 nd International Conference on Next Generation Computing Technologies*, Dehradun, 2016.
- [22] R Hawi, G Okeyo, M Kimwele, *Smart Traffic Light Control using Fuzzy Logic and Wireless Sensor Network*, in *Computing Conference*. London. 2017.

- [23] D. Hartanti, R. N. Aziza, and P. C. Siswipraptini, "Optimization of smart traffic lights to prevent traffic congestion using fuzzy logic," *TELKOMNIKA*, vol. 17, no. 1, pp. 320–327, 2019, doi:10.12928/TELKOMNIKA.v17i1.10129.
- [24] Y. Wei, Q. Shao, Y. Han, and B. Fan, "Intersection signal control approach based on pso and simulation," in *Second International Conference on Genetic and Evolutionary Computing*. IEEE, 2008, pp. 277–280.
- [25] E. Bingham, "Reinforcement learning in neurofuzzy traffic signal control," *European Journal of Operational Research*, vol. 131, no. 2, pp. 232–241, 2001.
- [26] X.F. CHEN et Z.-K. SHI. Real-coded genetic algorithm for signal timing optimization of a single intersection. In *International Conference on Machine Learning and Cybernetics*, volume 3, pages 1245 - 1248, 2002.
- [27] K. B. Kesur, "Advances in genetic algorithm optimization of traffic signals," *Journal of Transportation Engineering*, 2009.
- [28] T. L. Thorpe and C. W. Anderson, "Traffic light control using sarsa with three state representations," tech. rep., Citeseer, 1996.
- [29] M. Wiering et al., "Multi-agent reinforcement learning for traffic light control," in *ICML*, pp. 1151–1158, 2000.
- [30] M. Abdoos, N. Mozayani, and A. L. C. Bazzan, "Traffic light control in non-stationary environments based on multi agent Q-learning," in *14th International IEEE Conference on Intelligent Transportation Systems (ITSC)*, Washington, DC, USA., 2011, pp. 1580–1585.
- [30] K. Prabuchandran, H. K. AN, and S. Bhatnagar, "Multi-agent reinforcement learning for traffic signal control," in *Intelligent Transportation systems (ITSC), 2014 IEEE 17th International Conference on*. IEEE, 2014, pp. 2529–2534.
- [31] S. S. Mousavi, M. Schukat, and E. Howley, "Traffic light control using deep policy-gradient and value-function- based reinforcement learning," *IET Intell. Transp. Syst.*, vol. 11, pp. 417–423, 2017, doi:10.1049/iet-its.2017.0153.
- [32] Y.Liu, L.Liu, W.Chen. Intelligent Traffic Light Control Using Distributed Multi-agent Q Learning. *IEEE 20th International Conference on Intelligent Transportation System (ITSC)*. 2017.

- [33] W. Gender. Reinforcement Learning Traffic Signal Control. Thèse de doctorat. McMaster University. Hamilton, Ontario (Civil Engineering). 2018.
- [34] A. Aouaichia et O.Bouhalouffa, Apprentissage par renforcement pour l'ajustement automatique des régulateurs en boucle, Mémoire d'ingénieur, Ministry of higher education and scientific research Tlemcen, 2020.
- [35] Y.Zennir, Apprentissages par renforcement et systèmes distribués : Application à l'apprentissage de la marche d'un robot hexapode, Thèse de doctorat, E.E.A, Lyon , 2004.
- [36] A.Léon, Apprentissage séquentielle budgétisé pour la classification extrême et la découverte de hiérarchie en apprentissage par renforcement, Thèse de doctorat, Sorbone University, Paris, 2019.
- [37] D.L.S.Marta, Deep Learning Methods for Reinforcement Learning, Thesis to obtain the Master of Science Degree in Aerospace Engineering, 2016.
- [38] M.Benyamine, Deep Reinforcement Learning for Complex Manipulation Tasks with Sparse Feedback, A thesis submitted in fulfillment of the requirements for the degree of Master of Industrial Engineering and Management, Ben-g urion university of the negev, 2020
- [39] A.Vidali, L.Crociani, G.Vizzari, S.Bandini, A Deep Reinforcement Learning Approach to Adaptive Traffic Lights Management, University of Milano-Bicocca, Milano, Italy, 2019.
- [40] M.Boughaba et B.Boukhris, L'apprentissage profond (Deep Learning) pour la classification et la recherche d'images par le contenu, Mémoire master professionnel, Université Ksdi Merbah Ouargla, 2017.
- [41] J.Aurore, Modélisation et synthèse de voix chantée à partir de descripteurs visuels extraits d'images échographiques et optiques des articulateurs, Thèse de doctorat, University pierre et Marie Curie, 2016.